

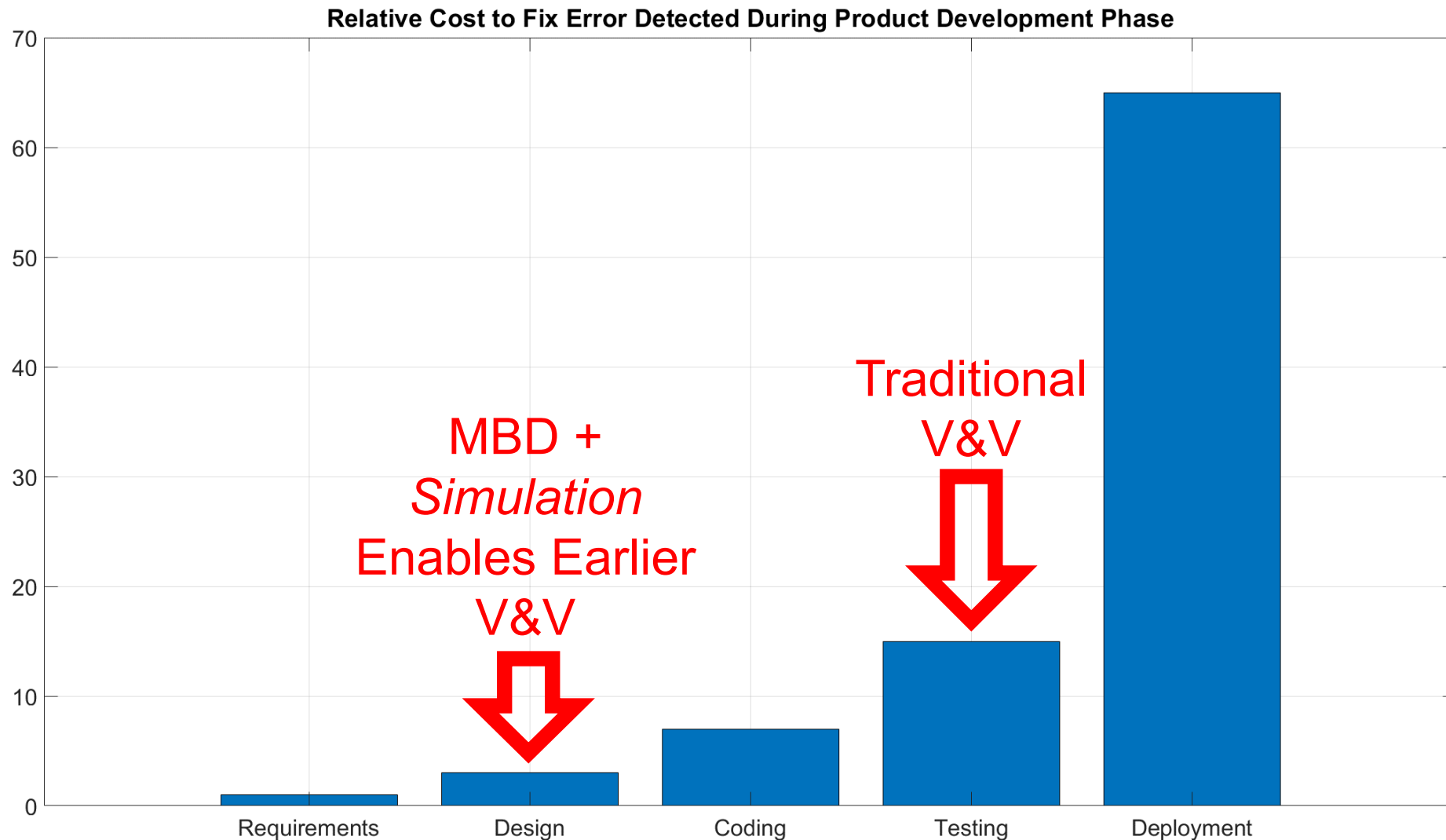
MATLAB EXPO

形式化需求和基于需求的测试用例生成

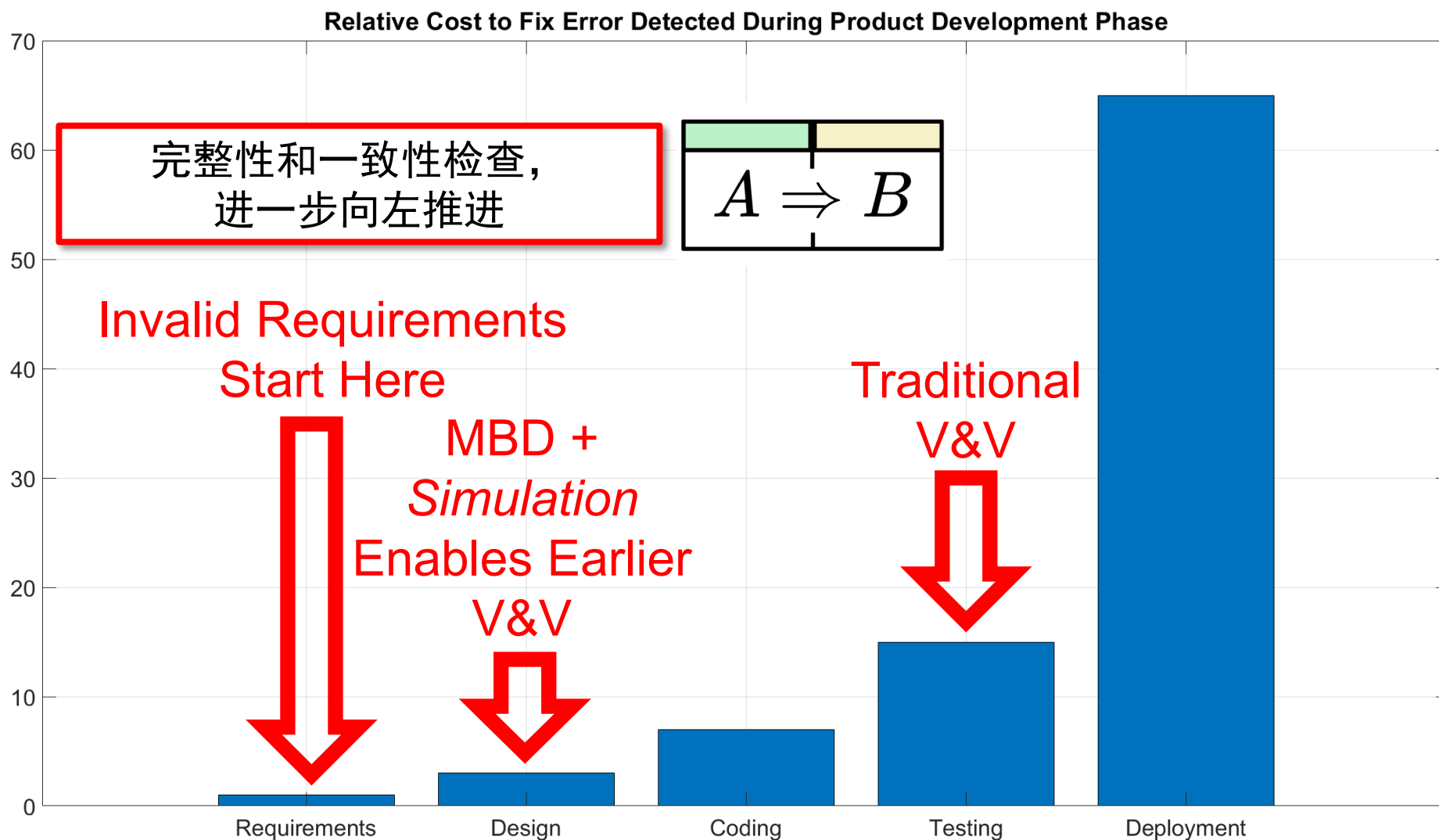
苏哲, MathWorks 中国



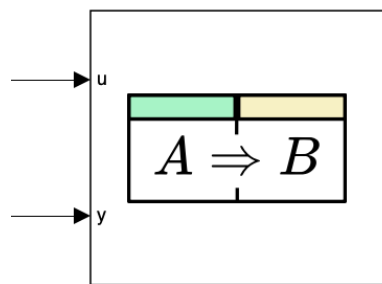
Model-Based Design: 早期发现错误以降低项目开销



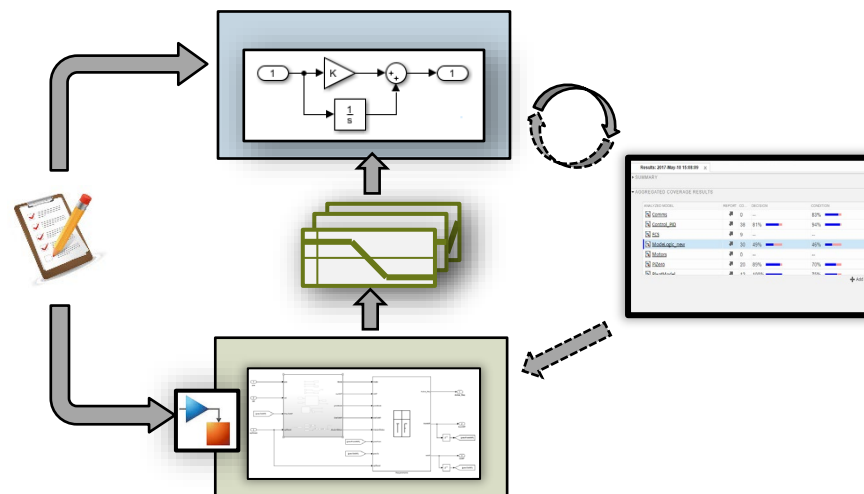
我们可以再早一点发现错误吗？



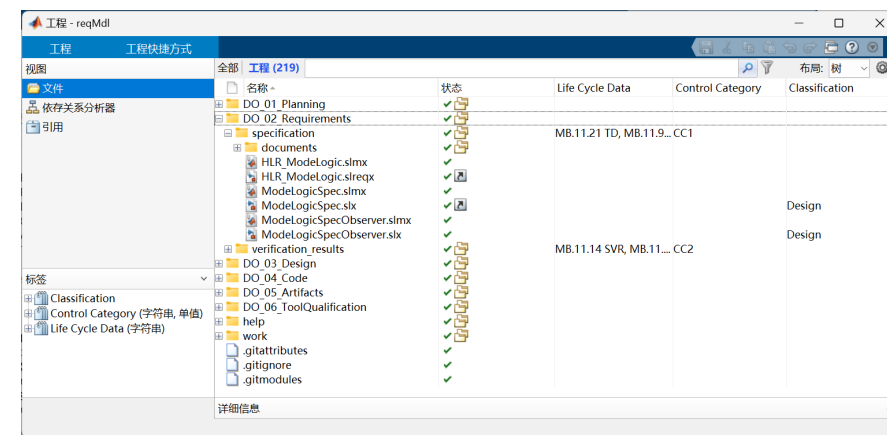
主要内容



形式化需求工具：
Requirements Table

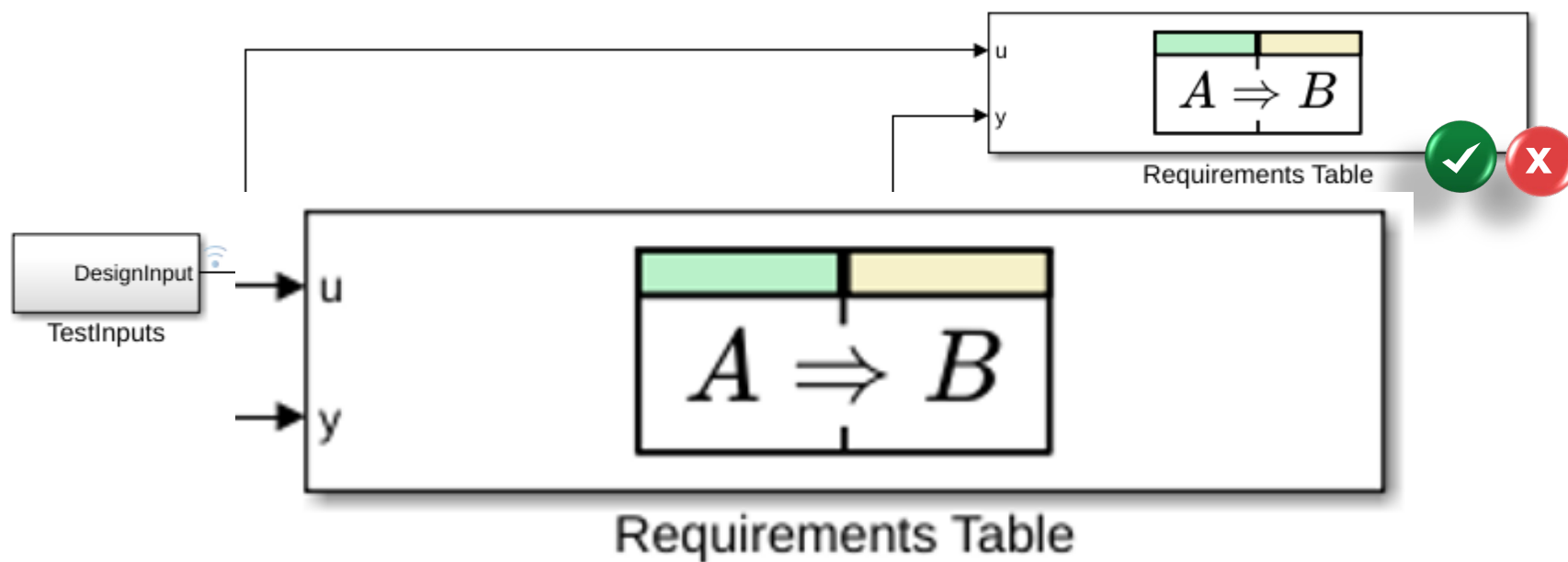


基于形式化需求的测试 workflow



workflow应用举例

Requirements Table 介绍



Requirements Table 基础语法

Precondition			Duration	Postcondition	
AP_Switch	Phi_TE	Roll_Ref_TN		RollMode	TRA_CMD
ON			3	HOLD	
<p>if (condition involving system inputs == true) for N seconds require that (condition involving system outputs == true)</p>					
	[-30, -6] [6, 30]				PHI

Requirements Table 支持多层次结构

Requirements		Assumptions					
Index	Summary	Precondition			Duration	Postcondition	
		AP_Switch	Phi_TE	Roll_Ref_TN		RollMode	TRA_CMD
1		ON			3	HOLD	
2		OFF				USR	
3				(-3, 3)			
3.1			[-6,6]				0
3.2			> 30				30
3.3			< -30				-30
3.4			[-30, -6] [6, 30]				PHI

- Multiple requirements can be active at the same time
- Sub-requirements are active if parent requirements are active

Requirements Table 快速查看需求

Requirements		Assumptions					
Index	Summary	Precondition			Duration	Postcondition	
		AP_Switch	Phi_TE	Roll_Ref_TN		RollMode	TRA_CMD
1		ON			3	HOLD	
2		OFF				USR	
3				(-3, 3)			
3.1			[-6,6]				0
3.2			> 30				30
3.3			< -30				-30
3.4			[-30, -6] [6, 30]				PHI

Requirements Table 支持简洁的数学表达式

Requirements		Assumptions					
Index	Summary	Precondition			Duration	Postcondition	
		AP_Switch	Phi_TE	Roll_Ref_TN		RollMode	TRA_CMD
1		ON			3	HOLD	
2		OFF				USR	
3				(-3, 3)			
3.1			[-6,6]				0
3.2			> 30				30
3.3			< -30				-30
3.4			[-30, -6) [6, 30]				PHI

Requirements Table 支持简洁的数学表达式

Requirements		Assumptions					
Index	Summary	Precondition			Duration	Postcondition	
		AP_Switch	Phi_TE	Roll_Ref_TN		RollMode	TRA_CMD
1		ON			3	HOLD	
2		OFF					
3				(-3, 3)			
3.1			[-6, 6]				0
3.2			> 30				30
3.3			< -30				-30
3.4			[-30, -6) [6, 30]				PHI

Phi_TE >= -6 && Phi_TE <= 6

Requirements Table 支持复杂的数学表达式或者内置的函数

Requirements		Assumptions			
Index	Summary	Precondition	Duration	Postcondition	
1		AP_Switch == ON	3	RollMode == HOLD	_CMD
2					
3					
3.1					0
3.2					30
3.3					-30
3.4		(Phi_TE >= 6 && ... Phi_TE < -6) ... (Phi_TE >= 6 && ... Phi_TE <= 30)			PHI

Builtin Temporal Operators:

- prev** : Value of an input at the last time step
- hasChanged**: Has the value changed since the last time step?

可以在Requirements Editor 中查看 Requirements Table

The screenshot displays the Requirements Editor interface. On the left, a 'Requirements Table' is visible with the following data:

Index	Summary
1	Roll Hold mode
2	Roll User Mode
3	Valid roll settings specifications
3.1	Zero setpoint detection
3.2	Positive saturation detection
3.3	Negative saturation detection
3.4	Nominal behavior

The main window shows a tree view of the project structure, with 'Table1' expanded to show the following table:

Index	ID	Summary
1	#6	Roll Hold mode
2	#7	Roll User Mode
3	#8	Valid roll settings specificatio...
3.1	#9	Zero setpoint detection
3.2	#10	Positive saturation detection
3.3	#11	Negative saturation detection
3.4	#12	Nominal behavior

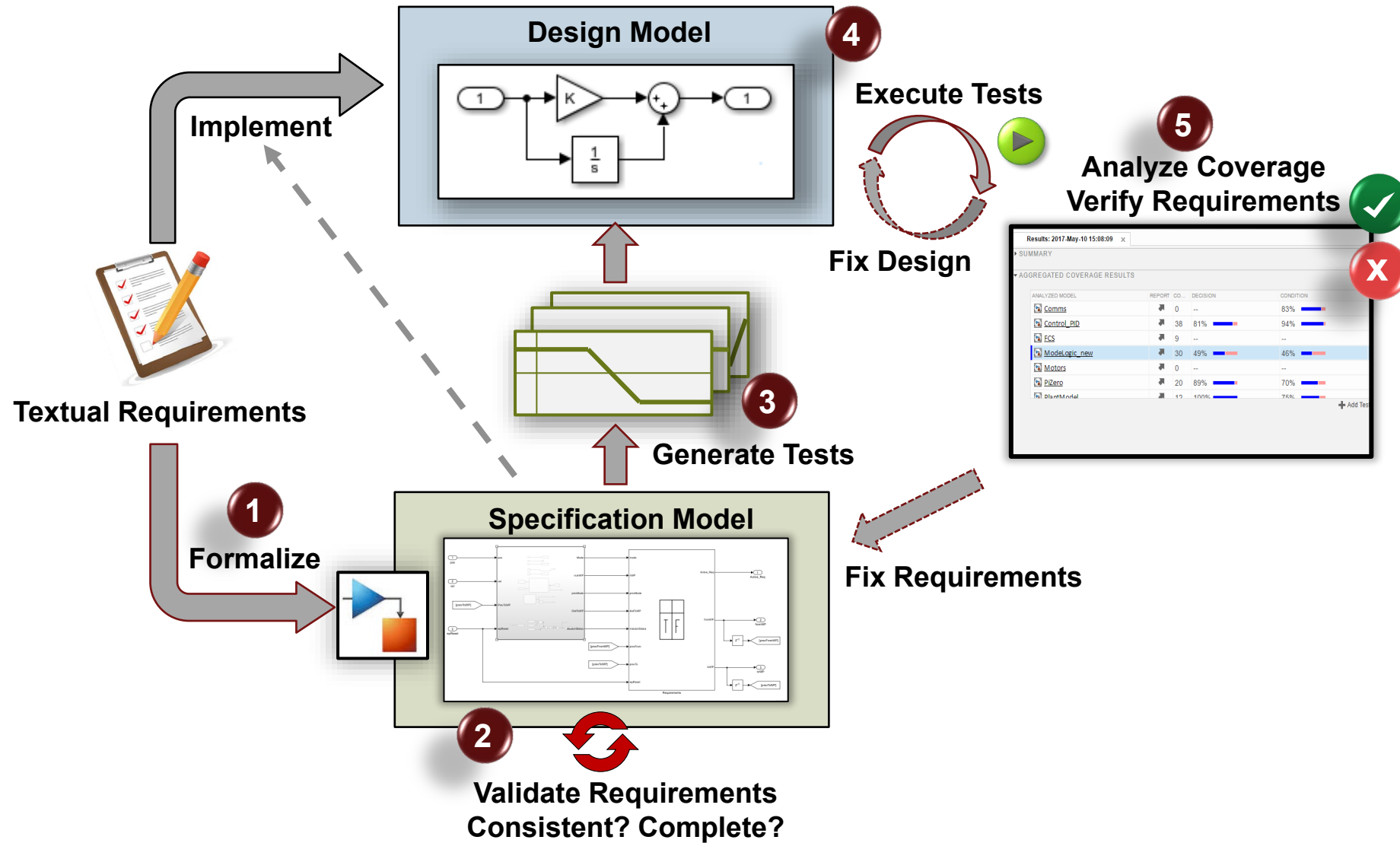
The right-hand pane shows the 'Properties' view for 'Requirement: #6'. The 'Type' is set to 'Functional'. The 'Index' is '1' and the 'Custom ID' is '#6'. The 'Summary' is 'Roll Hold mode'. The 'Description' tab is active, showing the following text:

Roll hold mode shall be the active mode whenever the autopilot is engaged and no other lateral mode is active.

The 'Rationale' tab is also visible. Below the description, there are fields for 'Keywords', 'Revision information', and 'Links'. The 'Links' section shows a link to 'IDLE'.

Overlaid text boxes highlight key features: 'Brief Descriptions' points to the table, 'Rich Text' points to the description field, and 'Links' points to the link field.

基于形式化需求的测试 workflow



基于形式化需求的测试应用举例

工程 - reqMdl

工程 工程快捷方式

视图 全部 工程 (219) 布局: 树

名称	状态	Life Cycle Data	Control Category	Classification
DO_01_Planning	✓			
DO_02_Requirements	✓			
specification	✓	MB.11.21 TD, MB.11.9...	CC1	
documents	✓			
HLR_ModelLogic.slmx	✓			
HLR_ModelLogic.slreqx	✓			
ModelLogicSpec.slmx	✓			
ModelLogicSpec.slx	✓			Design
ModelLogicSpecObserver.slmx	✓			
ModelLogicSpecObserver.slx	✓			Design
verification_results	✓	MB.11.14 SVR, MB.11....	CC2	
DO_03_Design	✓			
DO_04_Code	✓			
DO_05_Artifacts	✓			
DO_06_ToolQualification	✓			
help	✓			
work	✓			
.gitattributes	✓			
.gitignore	✓			
.gitmodules	✓			

标签

- Classification
- Control Category (字符串, 单值)
- Life Cycle Data (字符串)

依存关系分析器

引用

详细信息

应用举例：文本需求

The screenshot displays the Requirements Editor interface. On the left, a tree view shows the project structure with 'HLR_ModelLogic.sreqx' highlighted. The main window is divided into two panes. The left pane shows a table of requirements, and the right pane shows the details for 'Requirement: 17'.

Index	ID	Summary
> HLR_ModelLogic		
✓ HLR_ModelLogicFixed*		
1	3	Modes of Operation
2	#61	Transition Priority
3	16	Establish Communications
3.1	17	Enter Establish Communications at Power Up
3.2	18	Enter Establish Communications from Crash
3.3	19	Enter Establish Communications from Land
> 4	21	Initialization
> 5	25	Calibrate the Sensors
> 6	29	Ready for Flight
> 7	33	Track Altitude
> 8	37	Track Altitude and Position
> 9	41	Lost Ball
> 10	46	Land
> 11	50	Crash
> 12	#60	Invalid Transition

The right pane shows the details for 'Requirement: 17':

- Properties:** Type: Functional, Index: 3.1, Custom ID: 17, Summary: Enter Establish Communications at Power Up.
- Description:** MS UI Gothic, 10. The Establish Communications mode shall be entered upon power up of the flight control system.
- Keywords:**
- Revision information:**
- Links:** Implemented by: Default Transition 2, WaitForComms, Default Transition 2, WaitForComms.
- Comments:**

需求条目

需求详细内容

需求追溯

应用举例：形式化需求

Requirements Table: ModeLogicSpec/Requirements Table * - Simulink

仿真 调试 建模 格式 表 APP

工程 新建 打开 保存 打印 库浏览器 表属性 更新模型 分析表 停止时间 10.0 普通 快速重启 后退 运行 (覆盖率) 仿真

模型浏览器 Requirements Table

索引	摘要	预条件
1	true	getSimulationTime() == 0 getPrevious(Mode)
2	false	
2.1		WaitForComms GCSCmds.WIFIconnected && ... GCSCmds.BTconnected
2.2		Init GCSCmds.CalibrateCmd
2.3		Calibration GCSCmds.GCS_MissionMode == uint8(1) && ... State.CalibrationDone
2.4		ReadyForTO GCSCmds.CalibrateCmd
2.5		ReadyForTO IsBall
2.6		TrackAlt State.Altitude >= single(0.1)
2.7		TrackAlt ~IsBall
2.8		TrackAlt GCSCmds.GCS_MissionMode == uint8(2) ... State.BatteryVolts <= single(3)

就绪 84% FixedStepDiscrete

不一致性问题

不一致性 1: 对于以下输入, 需求 2.4 和 2.5 中的 'Mode' 在时间 0.4 不一致:

Time	0-0.1	0.2	0.3	0.4
Step	1-2	3	4	5
• GCSCmds.EMERGENCY_OFF	0	0	0	0
• GCSCmds.WIFIconnected	1	0	0	0
• GCSCmds.BTconnected	1	0	0	0
• GCSCmds.CalibrateCmd	0	1	0	1

不完整性问题

不完整性 1: 对于以下输入, 没有在时间 0.1 指定 'Mode':

Time	0	0.1
Step	1	2
• GCSCmds.EMERGENCY_OFF	0	0
• GCSCmds.WIFIconnected	1	0
• GCSCmds.BTconnected	1	0
• GCSCmds.CalibrateCmd	0	0
• GCSCmds.GCS_MissionMode	0	0

应用举例：生成基于需求的测试用例

The screenshot displays the Simulink Design Verifier interface for a project named 'ModeLogicSpec'. The 'Generate Tests' button is highlighted with a red box. The main workspace shows a 'Requirements Table' with the following entries:

ID	Requirement Name	Requirement Type	Target
1	GCSCmds	GCSCmds	GCSCmds
2	IsBall	IsBall	IsBall
3	State	State	State
4	Logic	State	<Mode>

The requirements table is represented by a diagram with a box containing the logical expression $A \Rightarrow B$. The status bar at the bottom indicates '就绪' (Ready) and '100%'.

Simulink Design Verifier Results Summary: ModeLogicSpec

Progress

Objectives processed	54/54
Satisfied	0
Unsatisfiable	0
Elapsed time	0:40

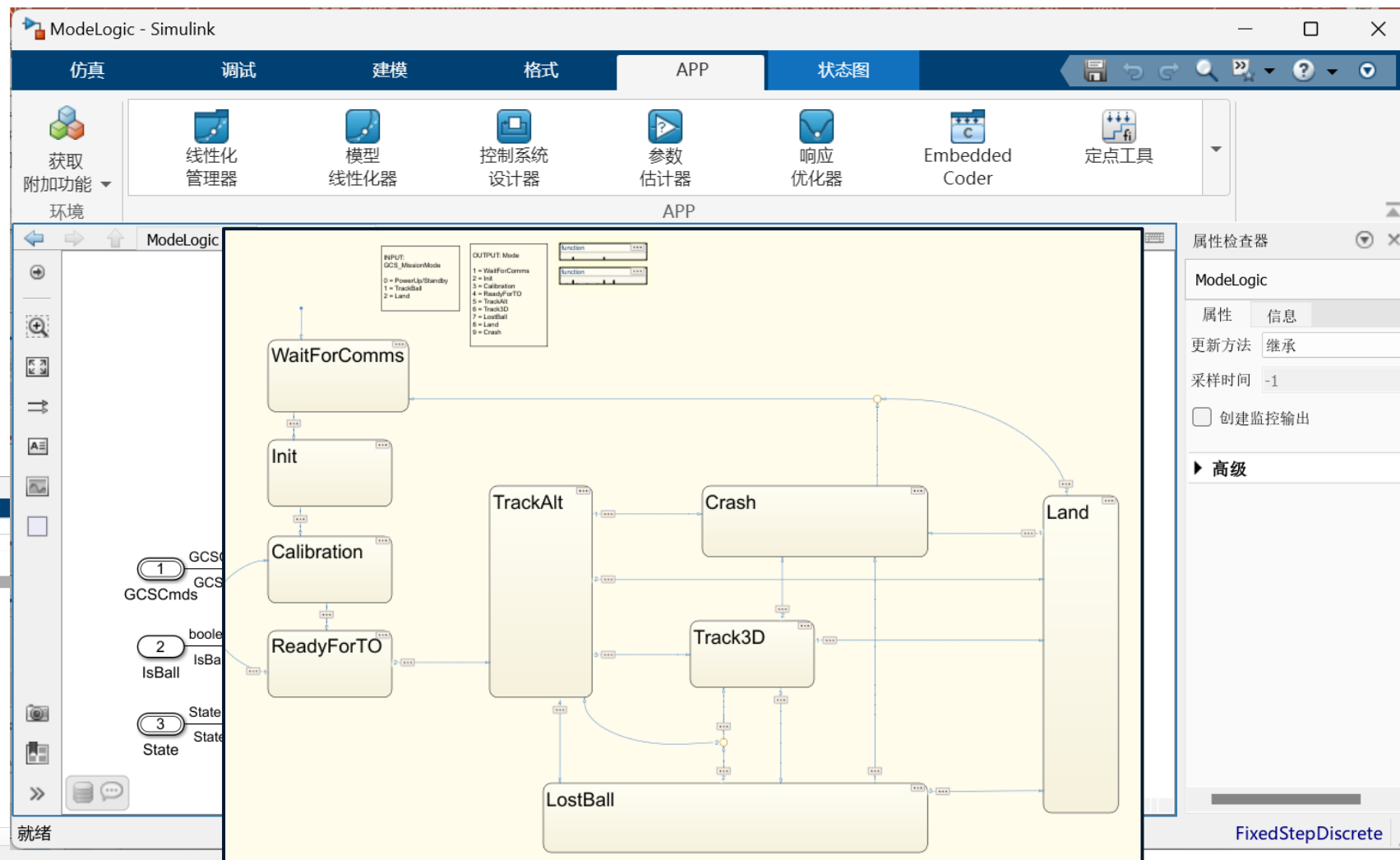
Test generation completed normally.
54/54 objectives undecided due to runtime error

Results:

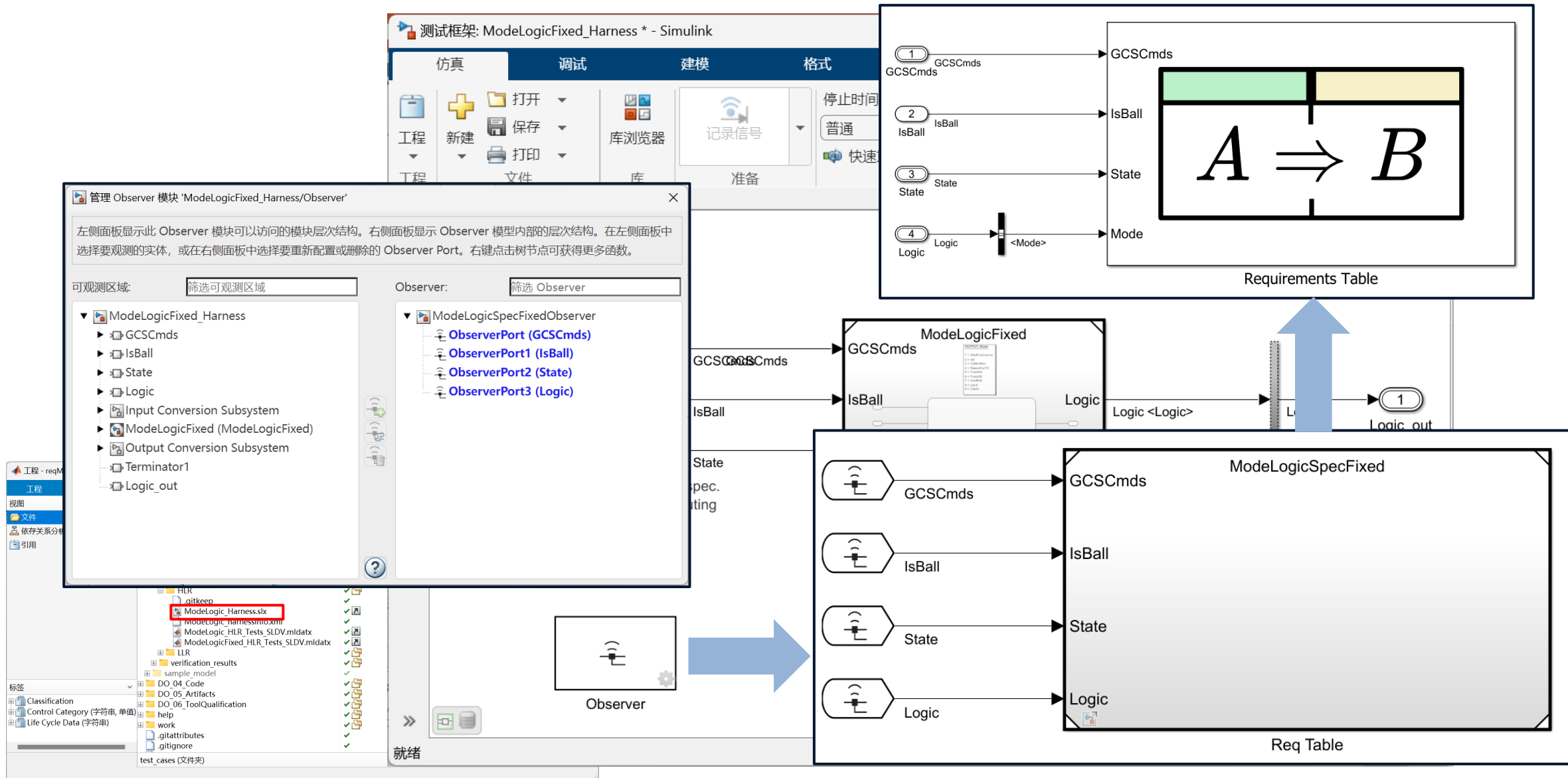
- [Open filter explorer](#)
- [Highlight analysis results on model](#)
- [View tests in Simulation Data Inspector](#)
- [Detailed analysis report: \(HTML\) \(PDF\)](#)
- [Create harness model](#)
- [Save test cases/counterexamples to spreadsheet](#)
- [Export test cases to Simulink Test](#)
- [Simulate tests and produce a model coverage report](#)

Data saved in: [ModeLogicSpec_sldvdata.mat](#)
in folder: [C:\works\RerMod\reqmdl-master\reqmdl-master\sldv_output\ModeLogicSpec](#)

应用举例：逻辑建模



应用举例：搭建 test harness



应用举例：在Simulink Test 中建立测试

The screenshot displays the Simulink Test Manager interface. On the left, a file browser shows the project structure, with the file `ModelLogic_Harness.six` highlighted. The main window is titled "TESTS" and shows the configuration for a test harness named "ModeLogic_Harness".

The "TESTS" window includes a "Test Browser" on the left, a "Filter tests by name or tags, e.g. tags: test" search bar, and a "Results and Artifacts" tab. The "Start Page" tab is active, showing the following configuration:

- Model: ModeLogic
- TEST HARNESS*
 - Harness: ModeLogic_Harness
- SIMULATION SETTINGS AND RELEASE OVERRIDES
- ITERATIONS*
 - TABLE ITERATIONS*

NAME	DESCRIPTION	EXTERNAL INPUT
<input checked="" type="checkbox"/> Iteration1	None	Test Case:1
<input checked="" type="checkbox"/> Iteration2	None	Test Case:2
<input checked="" type="checkbox"/> Iteration3	None	Test Case:3
<input checked="" type="checkbox"/> Iteration4	None	Test Case:4
<input checked="" type="checkbox"/> Iteration5	None	Test Case:5
<input checked="" type="checkbox"/> Iteration6	None	Test Case:6
 - SCRIPTED ITERATIONS
 - Run test iterations in fast restart

At the bottom of the "ITERATIONS*" section, there are buttons for "Auto Generate", "+ Add", and "Delete". A "Show Iterations" button is also present, with the tooltip "Show the list of iterations that will execute".

The "PROPERTY VALUE" table at the bottom left of the main window shows the following details for the selected test:

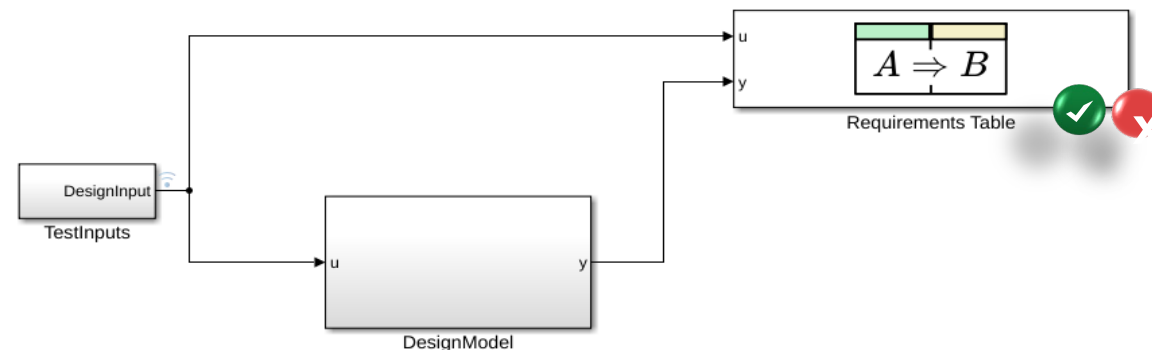
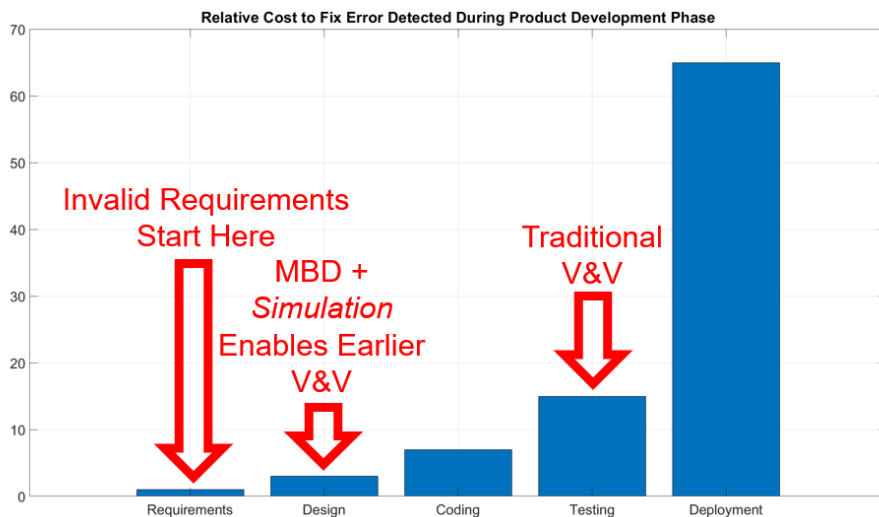
PROPERTY	VALUE
Name	Tests From Spec M
Type	Simulation Test
Model	ModeLogic
Harness Name	ModeLogic_Harness
Simulation Mode	[Model Settings]
Location	C:\works\RerModr...

应用举例：执行测试并查看测试结果

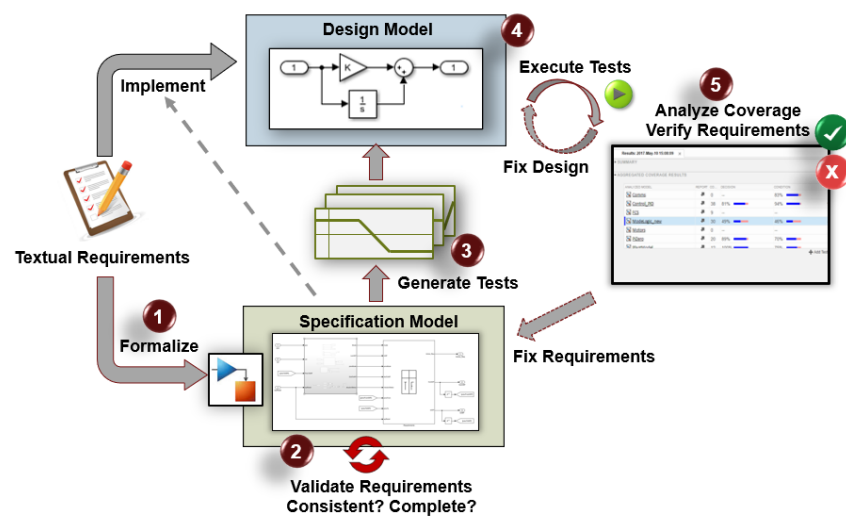
The screenshot displays the MATLAB Test Manager interface. The main window is titled "Test Manager" and features a toolbar with various icons for file operations (New, Open, Save, Cut, Copy, Paste, Delete), test execution (Run, Run with Stepper, Stop, Parallel), and reporting (Report, Visualize, Highlight in Model, Export, Model Testing Dashboard, Preferences, Help). Below the toolbar, the "Test Browser" tab is active, showing a table of test results. The table has columns for "NAME" and "STATUS". The results are grouped under "Results: 2023-May-13 17:20:55" and "Tests From Spec Model". The "Tests From Spec Model" group is expanded to show several test cases, including "Iteration1" through "Iteration5", "Verify Statements", "External Inputs", "GCSCmds", "IsBall", "State", "Logic", and "Sim Output (ModeLogic_Har)". The "State" test case is highlighted in blue. To the right of the test browser, a plot titled "IsBall" and "State.CalibrationDone" is shown. The plot has a y-axis from 0 to 1.0 and an x-axis from 0 to 1.1. The "IsBall" signal (red line) is a step function that jumps from 0 to 1 at approximately x=0.7. The "State.CalibrationDone" signal (orange line) is a step function that jumps from 0 to 1 at approximately x=0.7. Below the plot, a table shows the status of the test cases, with "ModeLogicSpecObserver" having a status of "100%".

NAME	STATUS
Results: 2023-May-13 17:20:55	34 ✓ 5 ✗
Tests From Spec Model	34 ✓ 5 ✗
Iteration1	✓
Iteration2	✓
Iteration3	✓
Iteration4	✗
Verify Statements	✗
External Inputs	
GCSCmds	
IsBall	
State	
Logic	
Sim Output (ModeLogic_Har)	
Iteration5	✓

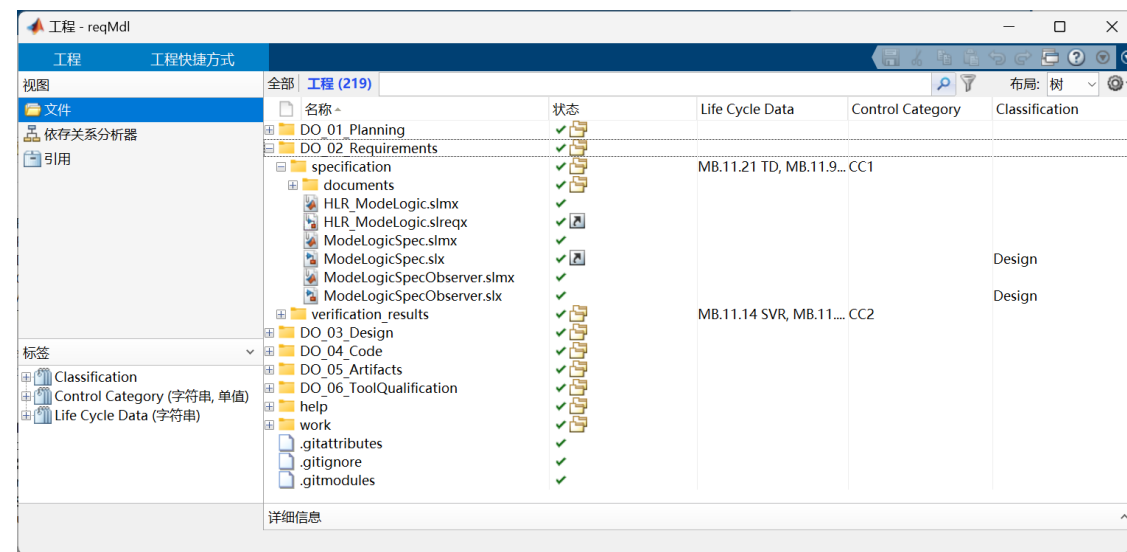
总结



形式化需求工具：Requirements Table



基于形式化需求的测试 workflow



workflow 应用举例

资源

产品主页

- [Requirements Toolbox - MATLAB & Simulink \(mathworks.com\)](https://www.mathworks.com/products/requirements.html)

文档：

- [Requirements Toolbox Documentation \(mathworks.com\)](https://www.mathworks.com/help/requirements/)
- [Requirements Definition \(mathworks.com\)](https://www.mathworks.com/help/requirements/requirements-definition.html)
- [Requirements Table Block \(mathworks.com\)](https://www.mathworks.com/help/requirements/requirements-table-block.html)
 - [Use a Requirements Table Block to Create Formal Requirements \(mathworks.com\)](https://www.mathworks.com/help/requirements/requirements-table-block-to-create-formal-requirements.html)

示例：

- [Requirements Toolbox — Examples \(mathworks.com\)](https://www.mathworks.com/help/requirements/examples.html)
- [Formalize Requirements in Simulink Models — Examples \(mathworks.com\)](https://www.mathworks.com/help/requirements/formalize-requirements-in-simulink-models-examples.html)

MATLAB EXPO

Thank you



© 2023 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [mathworks.com/trademarks](https://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.