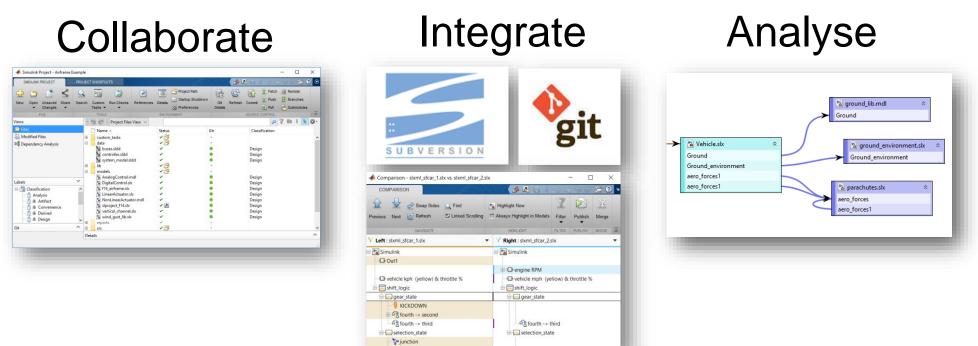
# MATLAB EXPO 2017

Team-Based Collaboration in Simulink

Sonia Bridge



Create tools that make it easy for teams to manage the full lifecycle of their Model-Based Design projects



kickdown



## **Common Challenges**

How to:

- Create a more efficient team-based environment?
- Effectively componentize system designs including data?
- Track design changes?
- Use source control functionality within Simulink?
- Associate project-level information with files?
- Utilise automation to maximise efficiency in enforcing best practices?
- Share work within the group and outside the group?
- Transfer knowledge across projects?



## **Simulink Project**

Foundation technology supporting efficient Model-Based Design in teams

- Enables sharing of work
- Ensures a project is complete (files, data)
- Ensures consistent environment across a team (MATLAB path, data, slprj location, ...)
- Supports different entry points and sharing best practices via shortcuts
- Enables associating project-level information with files via labels
- Part of Simulink (first shipped in R2011b)

Supports advanced users

- Impact analysis: what is the impact of a change?
- Rich APIs to script and customize

MATLAB EXPO 2017

SIMULINK PROJECT	PROJECT S	SHORTCUTS	DEPI	ENDENCY ANAL	YSIS		SA 1	XX		1 % B	i s	e 🗄 (	?
Open   Save As   Compare to Saved	Options Analyze			Horizontal		Select		è port ▼					
FILE	ANALYZE	VIEV	W	LAYO	UT	IMPA	CT ANALYSIS						
liews	· 5 ¢ ·								₽ 7			Impact Vie	w
Files     Modified Files	File Type 🌾							3	Dependency Typ	e Lift S	<b>ystem</b> C:\	\Refactore	ed
B Dependency Analysis												/09/2017 01	
					> 1	t_door_contro	ller.six			Drodu	icts (8)		
							da V		> 👔 lift_position_contro	MATI	AB 9.3		
						eneric_motor.s	ibt ©	_	Itt_position_contro		ink 9.0		
					/	$\sim$		1	> 📓 lift_intertia_utils.slx	Simso			
	SystemLevelTest.m	*	→ 👔 lift_system.s	dx ¥		t_inertia_pm.sl	x ¥	$\triangleleft$		Simso	ape Mult	ibody 5.1	
	SystemLevelTest.m	¥	→ 👔 lift_system.s	dx ¥		t_inertia_pm.sl	x \$	4	Ift_intertia_utils.six	Simso Simul	ape Mult ink Code		
< >>	(a) SystemLevelTest.m	¥.	→ 🛐 lift_system.s	dx ¥		t_inertia_pm.sl	x ¥	1		Simso Simul Show	ape Mult ink Code <u>All</u>	r 8.13	
	(A) SystemLevelTest.m	\$	→ 👔 lift_system.s	dx ¥				$\int$		Simsc Simul Show	ape Mult ink Code <u>All</u> oblems (	r 8.13 1)	
Labels 🗸	SystemLevelTest.m	*	→ 🕅 lift_system.s	dx ¥		it_inertia_pm.sl				Simsc Simul Show	ape Mult ink Code <u>All</u>	r 8.13 1)	
Labels 🗸	[]]SystemLeveTest.m	¥	▶ 🏠 lift_systems	aix ¥				$\left\{ \right\}$	A lift_intertiasidd	Simsc Simul Show	ape Mult ink Code <u>All</u> oblems (	r 8.13 1)	
Labels 👻	[A] Systemi.eveiTest.m	8	→ <u>I'a</u> lift_system.s	dr ¥]			Test.m ¥		Caller Internation	Simsc Simul Show	ape Mult ink Code <u>All</u> oblems (	r 8.13 1)	



# Getting started with an existing project

# Simulink project "mistake-proofs" your team environment

No more MATLAB code required to manage

- MATLAB Path via UI
- Locations for generated files ("slprj")

# *"I'm going to try my project on the new Linux cluster"*

Specify folders to add to the project path. Simulink Project adds these folders to the MATLAB search path when you open the project, and removes them when you close the project.

- 🚞 \$/CIUtils
- 🚞 \$/Components

Manage Project Path

- 🚞 \$/Components/Lift Door Controller
- 🚞 \$/Components/Lift Doors
- 🚞 \$/Components/Lift Inertia
- 🚞 \$/Components/Lift Position Controller



 $\times$ 



## **Simulink Projects Shortcuts**

- Make it easy for any engineer (not just the engineer who created the project) to:
  - Find important files
  - Find and execute important or common operations
    - Make the top-level model in the project a shortcut
  - All debuggable
- Optionally set tasks to run at project start-up or shutdown
  - Provides formal mechanism for running initialization scripts
  - Makes it easier to ensure the symmetric shutdown scripts are called



## **Task Automation – Configuring Project Environment**

📣 Simu	link Project	t - Lift System			- 🗆 X
SIMU	JLINK PROJ	ECT PROJE	ECT SHORTCUTS		i ti 🤊 🖨 🔁 📀 🖸
New Shortcut	Organize Groups	SetUpChangeSet	doors inertia		
MAN	AGE	GENERAL	SETUP TOP LEVEL FILES		
Views			Project Files View	1	₽ 7 🖿 I 🖁 @-
Config	uring Proj	ect Environment			
· - 승규	📄 Starti	ing Lift System			
<b>۲</b>		lize project path			
×		ng up working folders			
×		ify shadowed project	files		
× 1		s (\$\Data\doors.mat)			
Ľ.		a (\$\Data\inertia.mat			
<b>*</b> ./	🛃 setTe	amPrefs (\$\Utilities\s	et l'eamPrefs.m)		
Li					
Ŧ					
					Cancel
			Details		^

- Robustly configure the team environment
- For everyone
- Automatically



## Using Simulink Projects to Create a Consistent Cross-Team Environment

- Benefits:
  - Everyone on the team has the same environment
  - New team members can get started more quickly
  - Less wasted time debugging discrepancies



# **Integration with Source Control**



#### How do people share and manage projects?

At an SAE webinar on "Model-Based Engineering", question asked:

Q: "How do you manage the files and data within your projects?"

- 1. Named folders ("project\_v1", "project\_v2", etc.)
- 2. Source Control tool
- 3. Application Lifecycle Management (ALM) tool

#### How do people share and manage projects?

ALM

Named

Folders

Majority use COTS tools for managing work & sharing information

- Source control
- Application Lifecycle Management (ALM)

Surprise was the number just using the file system

- Doesn't scale well
- Doesn't support team work
- So why were they doing it?

Source

Control



#### **Source Control Integrations**

#### Microsoft Team Foundation Server (TFS) integration available <u>now</u> from MathWorks File Exchange



Products	Solutions	Academia	Support	Community	Events
File Ex	kchange	;			
TFS V	ersion Con	trol Integra	tion		
by Jaspe	e <mark>r Schneider</mark> 2016 (Updated 26	, in the second s			
	sion Control integ	ration in MATLAB	and Simulink		





#### **Compare and Merge Simulink Models**

# Simplified comparison and merge workflow for Simulink models

- Comparison and merge available with Simulink
- Easily select changes to merge into new target model file
- Highlight changes in the Simulink editor
- Launch comparison from the MATLAB desktop, current folder browser, command line, or source control
- Create reports for archiving and review

 $\gg$  slxml\_sfcar

MERGE       Image: State       Image:	📣 Comparison - slxml_sfcar_1.slx vs. slxml_sfcar_2.slx	-		×
Previous Next   Previous Next   Linked Sorolling   Bottom Model   V   V   V   V   Linked Sorolling   NAVIGATE   NAVIGATE   NAVIGATE   V   V   Linked Sorolling   Name   V   Linked Sorolling   V   V   Linked Sorolling   Name   V   Linked Sorolling   V   Name   V   Linked Sorolling   Linked Sorolling   Linked Sorolling   V   Linked Sorolling   Linked Sorolling   Linked Sorolling   Linked Sorolling   Name   Vehicle kph (yellow)& throttle %   Name   V   Linked Sorolling   Linked Soroll	MERGE	o e 1 e 1 - 5 ////////	20	• ?
Simulink     CD Out1     Name vehicle hph (yellow) & throttle %     CD Out1     CD Out1 <	Previous Next Grind Linked Scrolling	ways Highlight in Models Filter Save Close File Herge		4
-D Out1      -D Vehicle kph (yellow) & throttle %      Name vehicle kph (yellow) & throttle %      Name vehicle kph (yellow) & throttle %      Tengine: 1 -> transmission:1      T User Inputs: 2 -> Branch      Shift_logic	Y Left : slxml_sfcar_1.slx	<b>Right</b> : slxml_sfcar_2.slx		•
>D vehicle kph (yellow) & throttle %   Name   vehicle kph (yellow) & throttle %   Name   vehicle kph (yellow) & throttle %   Name   vehicle mph (yellow) & throttle %   Variation of the state of				^
Name vehicle kph (yellow)& throttle % <ul> <li>The Engine: 1 -&gt; transmission: 1</li> <li>The User Inputs: 2 -&gt; Branch</li> <li>The Shift_logic</li> <li>Target : targetFile.sk</li> <li>The The Simulink</li> <li>The Simulink</li> <li>The The Simulink</li> <li>The Si</li></ul>	- Versebiele keels (vellevu) 81 threattle 9/	-		
TE Engine: 1 -> transmission:1   Tuser Inputs:2 -> Branch   Shift_logic   gear_state   Simulink   Simulink   Outt   Simulink   Outt   Outt <td></td> <td></td> <td></td> <td></td>				
<ul> <li>Shift_logic</li> <li></li></ul>	Engine:1 -> transmission:1	Name vence npn genowice motile /s		
→ Shift_logic   → gear_state      Y Target : targetFile.slx Y Y Y Target : targetFile.slx Y Y Simulink O Outh O Out				
Y Target : targetFile.slx Y Y   Simulink O O   O-Out1 O O   O-Out	shift_logic	· · ·		
Y Target : targetFile.slx       Y       Y         Image: Simulink       Image: Simulink       Image: Simulink       Image: Simulink         Image: Image: Simulink       Image: Simulink       Image: Simulink       Image: Simulink         Image: Image: Simulink       Image: S	gear_state	gear_state		
Image: Simulink       Image: Simulink         Image: Outline       Image: Outline         Image: Simulink       Image: Outline         Image: Outline	🖗 KICKDOWN			
Outl-       ○       ○         Image: PM       ○       Image: PM         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice Phice mph (yellow) & throttle %         Image: Phice mph (yellow) & throttle %       ○       Image: Phice	Y Target : targetFile.slx		Y	Y
Image: PPM       Image: PPM         Image: Vehicle mph (yellow) & throttle %       Image: PPM         Image: Name_vehicle mph (yellow) & throttle %       Image: PPM         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Image: Problem ph (yellow) & throttle %         Image: Problem ph (yellow) & throttle %       Imag	🖃 🎦 Simulink			1
→□ vehicle mph (yellow) & throttle %       ○         Name       vehicle mph (yellow) & throttle %       ○         □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □			0	•
Name       vehicle mph (yellow)& throttle %       ○       ○         □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □       ○       ○       ○         □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □       ○       ○       ○         □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □			0	•
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □				
□-□ shift_logic □-□ gear_state			0	_
□-□ shift_logic □-□ gear_state	-		0	
□-□ shift_logic □-□ gear_state			õ	
□- □ shift_logic □- □ gear_state			ŏ	
			Ŭ	Ŭ
- § <del>kickdown</del>	ear_state			
				•



## **Integrating Work from Different Engineers via Merge**

Simulink Project - Elevator Systems SIMULINK PROJECT	PROJECT SHORTCUTS				
lew Open Share Check	Preferences     Project Path     References     ENVIRONMENT     TOOLS				
Price Oriento Price Price Pric	Name         Name         BatchJobs         Components         Lift Door Controller         Lift Doors         Lift Inertia         Lift Position Controller         Lift Supervisory Controller         Motors         Visualization         SystemLevelModels         Tests         Utilities	SVN SVN O O O O O O O O O O O O O	Classification None Design Design None Design Design Design Design Design None None Test None	Status · 엽 · 연 · 가 · 가 · 가 · 가 · 가 · 가 · 가 · 가	P     P     E     E       5     28       2       13       3       28       27       13       28       27       13       2       21       13       2       13       2       13       13       2       13       13       13       13       13       13       13       13       14
Labels	<u>∧</u>				
	Details				

- Supports concurrent
   engineering
- Lets you concentrate on design

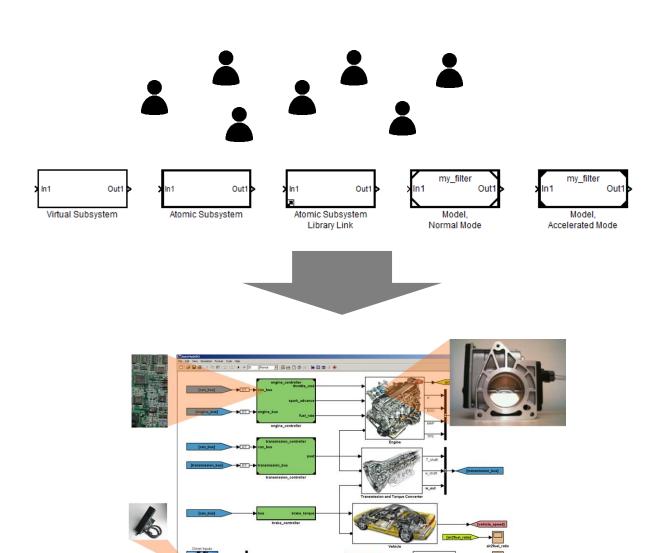


# Componentization



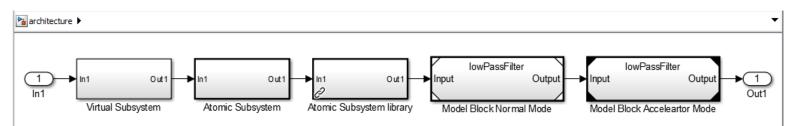
## **Complex Design Development through Componentization**

- Supporting team-based workflows
  - Faster modular development
  - More effective verification
  - Increased reusability





#### **Simulink Architectural Components**



- Virtual subsystem
  - Graphical component The contents are flattened to the level of the parent system during execution.
- Atomic subsystem
  - Simulink executes all blocks as a unit before executing the next block
  - Context dependent so inherits properties such as dimensions and data types from the parent model
- Model block
  - Executed as a unit

Context independent so doesn't inherit properties from parent model
 MATLAB EXPO 2017



#### **Component selection strategy**

- Virtual and Atomic Subsystems
  - When scalability is not an issue
  - When the atomic subsystem boundary is acceptable
  - During early development of the system
- Model Reference
  - When scalability is needed
  - When hard interfaces are critical
  - To enable concurrent teamwork and unit testing
- Library Components
  - Reused utility functions



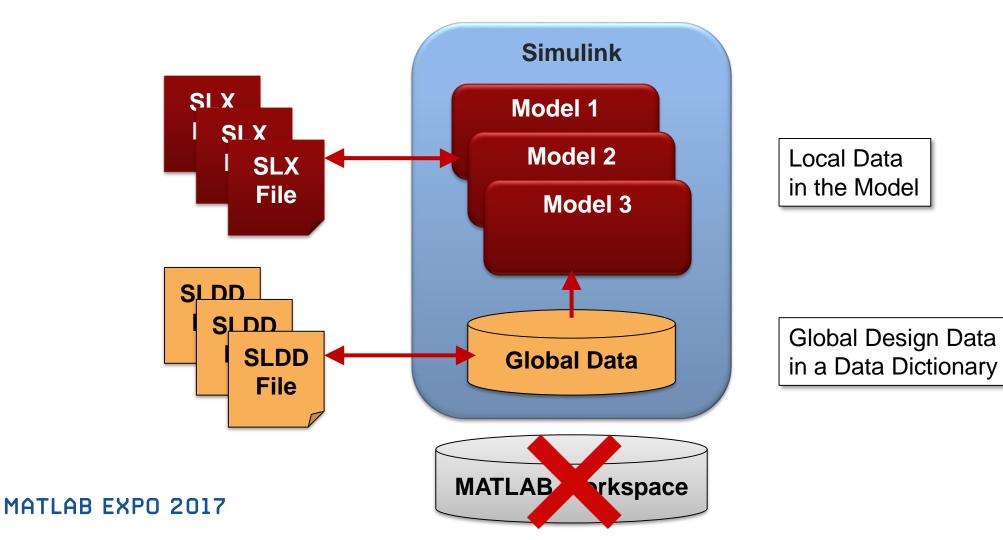
#### **Component-Based Modelling**

- Criteria for componentization:
  - Base the component boundaries on those of the real system
  - Define components distinctly so that only one engineer at a time needs to edit a component.
  - Subdivide components that are too big and those that could become too big as the design is elaborated.
- Recognize that there is no silver bullet
  - Experience is key here as well
- Start discussing this early in your project
  - What should be the criteria for componentization?
  - Who owns which component?



#### **Partitioning Design Data**

## Executable Specification = Algorithm + **Data**





#### Why Simulink Data Dictionary?

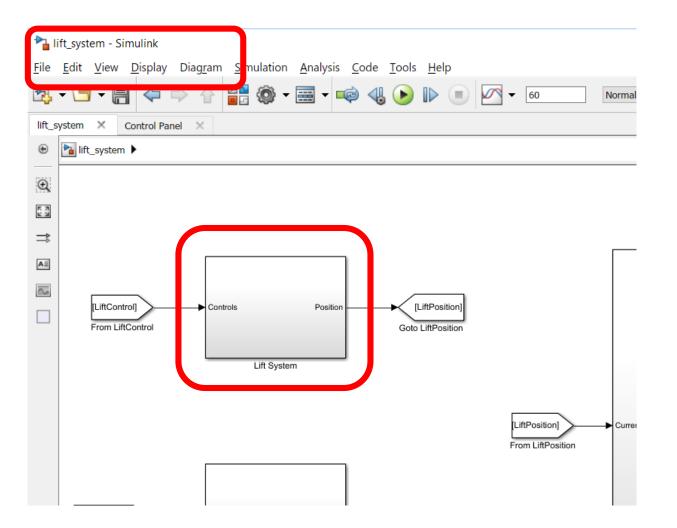
#### **Base Workspace Limitations**

- Mixed with MATLAB data
- Lack of organization
- Lack of change detection
  - What changed?
  - How did it change?
  - Can't revert changes
- Where did it come from?
- Lack of data persistence
- Conflict resolution issues

#### **Simulink Data Dictionary**

- Separate
- Partitioning
- Change detection
  - Shows changed items
  - Differencing
  - Revert
- Traceability
- Data persistence
- Conflict resolution





- This subsystem has same name as parent model
- Probably not the best name
- What is it..?



#### Demo

- Refactor into a new Model Reference
- Advisor helps automate/mistake proof the process
- Dependency analysis helps ensure we do not "lose" this new component
- Refactoring support for renaming
- Find dependencies to help work out why there are some other components with poorly chosen names (like "lift\_intertia")



## Simulink Project: Automatic Renaming

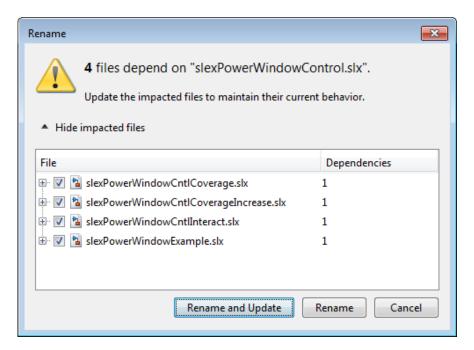
Automatically update files impacted by renaming, removing and deleting project files.

- Update model references and library links when renaming Simulink models.
- Update MATLAB code and model/block callbacks when renaming m/mlx files.
- Warn when deleting a file that is used by other files in the project.
- Update the MATLAB path when adding models or code files to the project.

>> slexPowerWindowStart

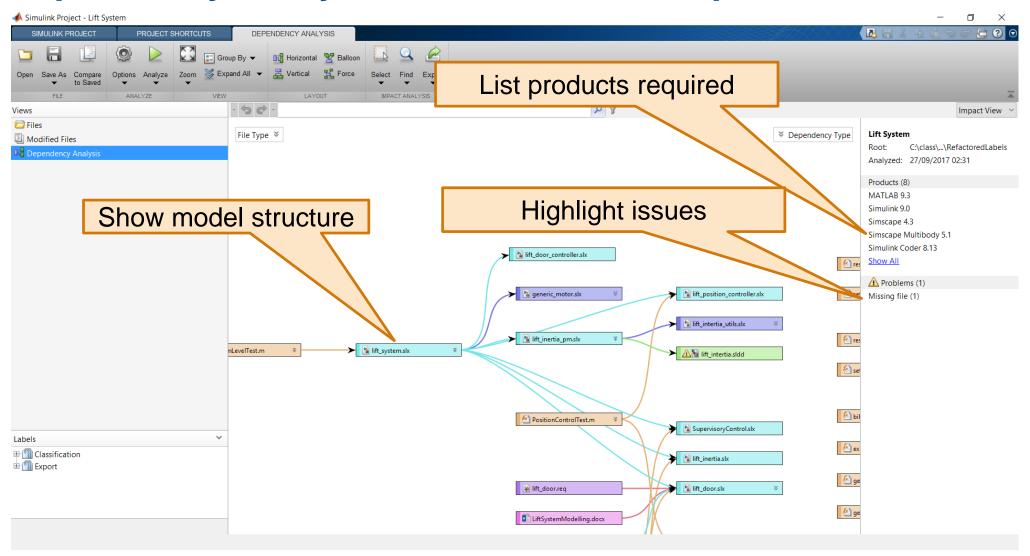
Rename, remove or delete a file from the project.

+		data
+		hmi
-		model
		🚡 slexPowerWindowCntlCoverage.slx
		🚡 slexPowerWindowCntlCoverageIncrease.slx
		🚡 slexPowerWindowCntlInteract.slx
		la slexPowerWindowControl.slx
		🖄 slexPowerWindowDSPLib.slx
		🚡 slexPowerWindowExample.slx
		🖄 slexPowerWindowPHYSLib.slx
	_	la slexPowerWindowSALib.slx





#### **Dependency Analysis – Modular Development**

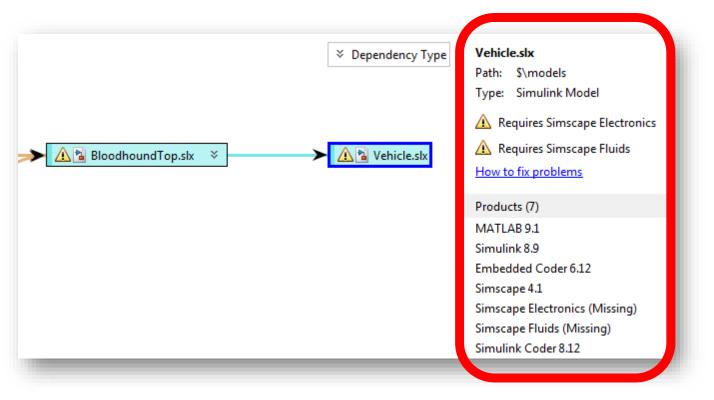




## Highlight Missing Products Required by a Project

# Find the products needed to use a project

- Dependency analysis reports the products required by a project
- Products that are not installed shown as "(Missing)".
- Files that use missing products show a warning icon. Click the file to see the missing products in the side panel.
- Open the model to get links to download missing products





# Using labels to share and store information



## Using Labels to Add Information to the Project

- Done lots of work to understand what the different parts are
- Wouldn't it be nice to record that so others do not have to repeat this?
- What are labels?
- Apply some labels to the project



#### **Simulink Project Labels**

#### Easily add, modify and view attached to a file.

- Easily see and edit label data attached to a file.
- Use drag and drop to add lab
- Easily switch between single-

	• 🕤 🖝 • Project	Files View 🔻				🔎 ү 🖿 i 🖪 @
nd view labels	🗋 Name 🔺	Status	SVN	Revision	Classification	Engineers
	🗄 🌗 data	✓ Ğ	•	2	None	
	🗉 퉲 models	< 4	•	2	None	
	🚡 AnalogCo	×	0	2	Design	Bob
	DigitalCon	×	•	2	Design	
hal data far all lahala	🕍 f14_airfra	×.	•	2	Design	
bel data for all labels	LinearActu			2	Design	
	NonLinear		•	2	Design	
	AnalogControl.mdl (Sir	mulink Model)				2 labels
add labels.	Model version :1.21		Engineers:			
	Saved in Simulink	version: R20	Bob	×		
	Last modified by: T	he MathWorl	Bob is devel	loping a		
single-valued labels.			new control			
0	(no description available)		Apply Cancel			
			Classification:			
			Design 🝷 🛛			
			Drag labels here			
Design 👻 🛛		F				
Utility here						
🖞 Derived						
🖞 Other						
8 Design						
None						
( <sup>®</sup> ) Test						
Convenience						
Artifact						



#### **Note on Metadata**

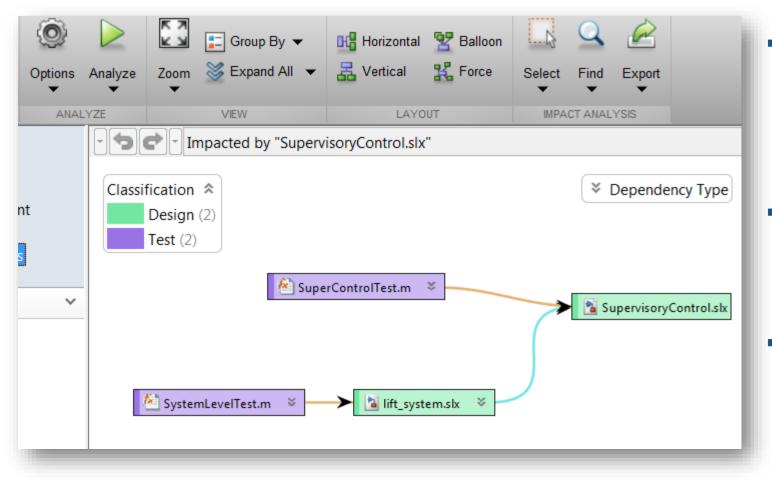
- What do we mean by metadata?
  - Wikipedia: "Data about data"
  - MathWorks: "Data about files"



- Data that is about the file, not (necessarily) part of it. For example:
  - FuelType = Gas, Diesel
  - ReleaseStatus = Research, Prototype, Production, Sunset
  - SecurityClassification = Unclassified, Protected, Restricted, Confidential
  - FileClassification = Design, Derived, Artefact
  - TestedWith = R2010b, R2011a, R2011b, ...
  - Coverage Metric = 84%
- Metadata can change without the file it relates to *having* to change.



#### Labels + Dependency analysis = Impact Analysis



- "What is the impact of changing the supervisory control model?"
  - "What tests do I need to run to verify those changes?"
- All accessible from command-line API for full automation



# More options for automation



#### Why Automate?

- Automated Processes Get Done
  - Regularly (if needed)
  - Repeatable
  - Can be done by anyone
- Corollaries
  - Manual processes are often infrequently done
  - Can be subject to variation
  - Perhaps only one person can do them



### How can Automation in Simulink Project help?

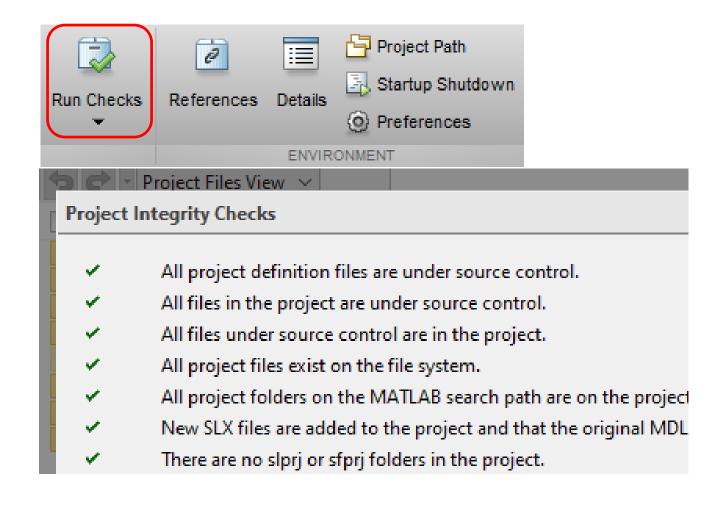
- Now anyone can rebuild the S-Functions
  - (or run the tests; generate code; publish the reports; import and validate test data; ...)
  - Even at 8:34pm on a Friday night; on a testing trip; ...
- Groups help provide structure
  - Group by type; or by job function (project manager group; testing group)

📣 Simulink Project - Airframe Example								
SIMU	LINK PROJE	CT PROJECT SHORT	CUTS					
New Shortcut	Organize Groups	System Design Description	F14 Model	Rebuild Project's S-functions				
MANA	AGE	DOCUMENTS	TOP LEVEL MODELS	UTILITY				



## **Automation Options in Simulink Projects**

- Build-in "best practice" support
  - Project Checks
  - Growing list of our own "gotchas"





#### **Run Custom Tasks and Create Reports**

#### Open custom task control from the toolstrip

- Select custom functions and files more easily
- View sets of results side-by-side
- Generate reports from custom task results

SI	MULINK	PROJECT		PROJEC	T SHORT	UTS				
New	Open	Unsaved Changes	Share	Search	Custom Task	Run Checks	References	Details Details Project Path O Preferences		
	FILE				TOOLS			ENVIRONMENT		

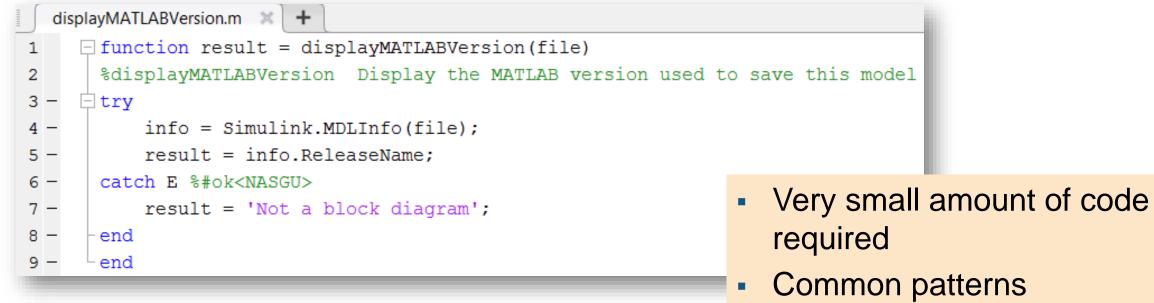
 Note: Custom tasks were known as "batch jobs" in releases before R2017a

	JLINK PROJECT		PROJECT SHORTCU	UTS		and the second				400			1 3 6 6
0	Dpen Unsaver Change		Search Custom Task	Run Checks	References	<ul> <li>Details</li> <li>Project Path</li> <li>Preferences</li> </ul>	SVN Details	C Refresh	Commit	Update	🖃 Show Log 🏐 Revert Project 父 Clear Login	SVN Cleanu Tag	p
	FILE		TOOLS	10-10-101	ENVI	RONMENT				SOURCE C	ONTROL		
us	tom Task												
Se	elect or create a	custom	ask to run on the in	cluded projec	t files.								
												Q	7 ⊻ @-
	📄 Name 🔺				Path					Include			
	AnalogC		📣 Custom Task R	Report - check	CodeProbler	ns						<b>- 0 - X</b>	<u>^</u>
	<ul> <li>buses.sldd</li> <li>checkCodeProble</li> <li>clean_up_project.</li> <li>controller.sldd</li> <li>DigitalControl.slx</li> <li>fl4 airframe.slx</li> </ul>	Files ran: 25 Completed at: 05/12/2016 10:57						<u>م</u> آ ه.			-		
		f14_airframe_test.r find_top_models.r LinearActuator.slx	🗋 Name 🔺			Path				Results			
			AnalogCon			\$\models \$\batch_jobs				1 problem	detected		H
	And and a second second second		billOfMater	rials.m	\$\batch_jobs \$\data				No problems detected	E			
	readme.txt rebuild_s_functior runUnitTest.m set_up_project.m slproject_f14.pdf			CheckCodeProblems.m						No proble	ms detected		
L			<ul> <li>clean_up_project.m</li> <li>controller.sldd</li> <li>DigitalControl.slx</li> <li>f14_airframe.slx</li> </ul>			S\utilities S\data S\models S\models			No problems detected				
	system_r		Results: analyzeMo	odelFiles.m								~	
			1 problem detected										
	timesthr												
	timesthro												Create
Cu	presents 100												
Cı	timesthro												

» sldemo\_slproject\_batchjobs
MATLAB EXPO 2017



#### **Example Custom Task**



- **Common patterns**
- Is this a file of type X?
- Does this file have a label from category X with value Y?



## **Simulink Project API**

```
X
 Command Window
                                                                         1
  Trial>> proj = simulinkproject
  proj =
    ProjectManager with properties:
                   Name: 'Lift System'
            Information: [1×1 slproject.Information]
           Dependencies: [1×1 slproject.Dependencies]
              Shortcuts: [1×6 slproject.Shortcut]
            ProjectPath: [1×6 slproject.PathFolder]
      ProjectReferences: [1×0 slproject.ProjectReference]
             Categories: [1×1 slproject.Category]
                  Files: [1×41 slproject.ProjectFile]
             RootFolder: 'C:\work\EXPO\ReadyToRefactor2'
  Trial>> proj.Files(12)
  ans =
    ProjectFile with properties:
                     Path: 'C:\work\EXPO\ReadyToRefactor2\Data\doors.mat'
                   Labels: [1×1 slproject.Label]
                 Revision: '3'
      SourceControlStatus: Unmodified
fx Trial>>
```

- Easily access information for the project
- Add, remove, inspect files and labels

If under source control,

- See source control information for files
- Get the list of modified files



# More options for sharing

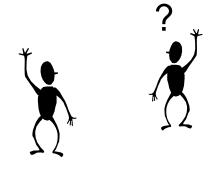


## Most Common Challenge in Sharing Work

"It works on my computer, just not on yours..."

Common causes:

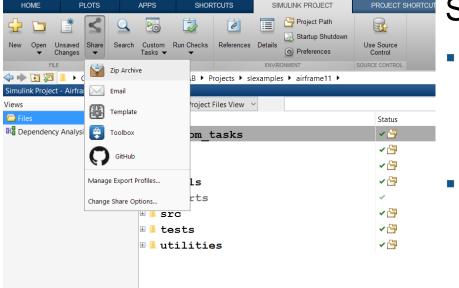
- Incomplete set of files
- Different environment
  - (software versions, MATLAB path, ...)
- Wrong data loaded
- What do I do to get started?





#### **Sharing work outside source control**

📣 MATLAB R2017b



Simulink Project has built in capabilities for sharing

- GitHub
  - Collaborative sharing
  - Expect to make changes together
- Archive file
  - Fast sharing of "what I am doing now"
  - "Delivery" workflows:
    - Send a package of work
    - Work independently
    - Receive a package of work back



#### How much to share?

- Typically do not want to share all my project with a supplier or customer
- Reduce to the minimum to
  - Avoid sharing IP I want to keep in-house
  - Keep it simple
- Create "Export Profiles" to manage which files are exported from project
  - Uses project labels to set up exclusion rules
  - Set up many profiles for different workflows
    - Sharing to supplier (share only what is needed)
    - Share to customer (shield my IP)
    - Share to HIL rig (no tests, doc, requirements)

• Etc. MATLAB EXPO 2017

Manage Export Profiles						
Controls Supplier Inc. HIL Rig	Export Profile: Controls Supplier Inc.					
	Files: Do not export files with these labels					
	🖞 Classification: Artifact 🛛 🖄 Classification: Derived					
	Labels: Do not export these labels					
+ -	🖞 Engineer: ffermani 🛛 🖞 Engineer: gwalker 🗶 🖞					



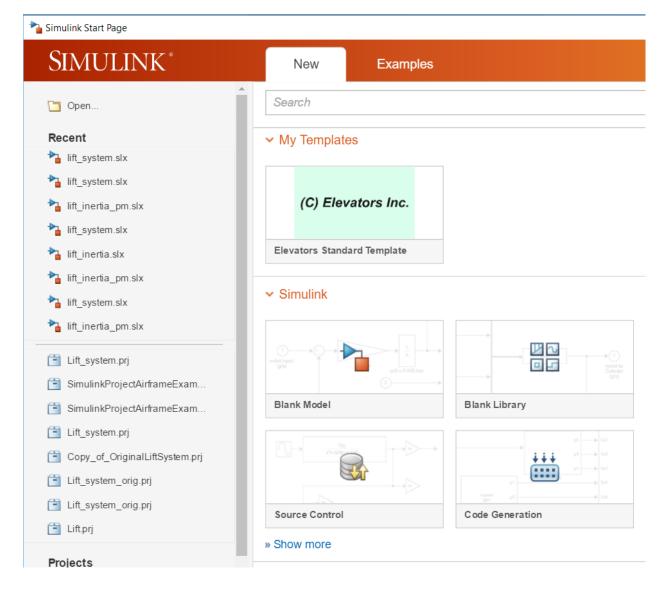
# **Knowledge transfer**



## **Model Templates**

Build models using design patterns that serve as starting points to solve common problems

- Use shipped templates to get started with building models or create custom templates to from a Simulink model
  - Avoid problem of corrupting original file when creating a new model
- Avoid repetitive tasks when starting out to build a new model
- Enforce a standard process for building models for the entire team or organization





#### **Projects can reference other projects**

# Componentize large modelling projects

- Develop reusable components using projects
- Flexible referencing:
  - Relative
  - Absolute
- Extract folders to referenced projects
- Deep hierarchies are supported

📣 Simu	link Project	- Airframe	Example
--------	--------------	------------	---------

SIMULINK PRO	PROJECT SHOR	TCUTS				ALL XX	
New Organize Shortcut	📆 Syster	m Design Description	📔 F14 Moo	del 📃	Signal Multiplier 👻 Wind Gust Library 👻		
MANAGE	MANAGE DOCUMENTS			ELS REF	ERENCED PROJECTS		
Views		- 5 6	Project Fi	les View	$\sim$		
🗁 Files		🗋 Nan	ne 🔺		Status	SVN	Revis
Modified Files	🗄 📒 cu	stom ta	asks	✓ <sup>1</sup> /2 <sup>2</sup>	•	2	
Dependency An	H Dependency Analysis				~ 🔄 ~ 🔄		2
							2

	Iles Modified Filer	Na	me 🔺	Status	SVN	Revision	Classification			
۲ ۲	Referenced Projects						×			
	<ul> <li>Signal Multiplier</li> <li>Wind Gust Library</li> </ul>	Relative reference:/refs/Wind Gust Library								
		H buses	Image: wind_gust_lib							
		DATA	TOP LEVEL MODEL							
Image: Wind Gust Library         Description:         This project provides a library which models the Dryden wind gust.										
suilt S R	+ • -	Project ro	ot: sey\AppData\Lo	ocal\Temp\slexamples	\airRef3\refs\Wind Gust	Library Set As Curren	nt Folder			
og	sion 1.8.						Close			

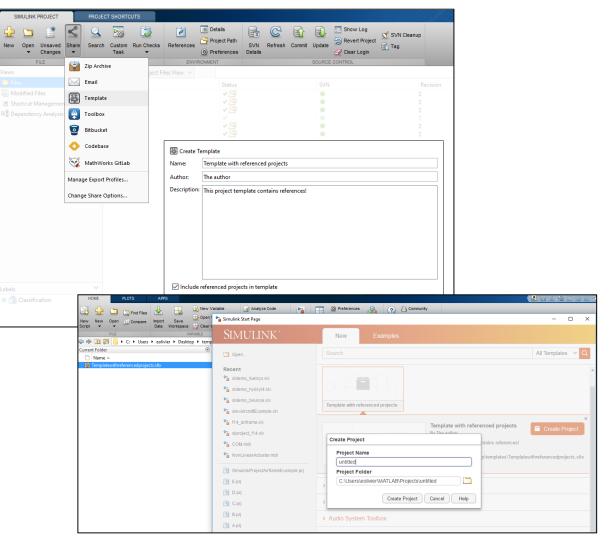
>> sldemo\_slproject\_airframe\_references



# Include References in Templates for Sharing and Reuse

#### **Template with references**

- Start from a project with references
- Create a template including the references
- Save it on the MATLAB path or double click it to see it in the start page
- Create a new project based on the template





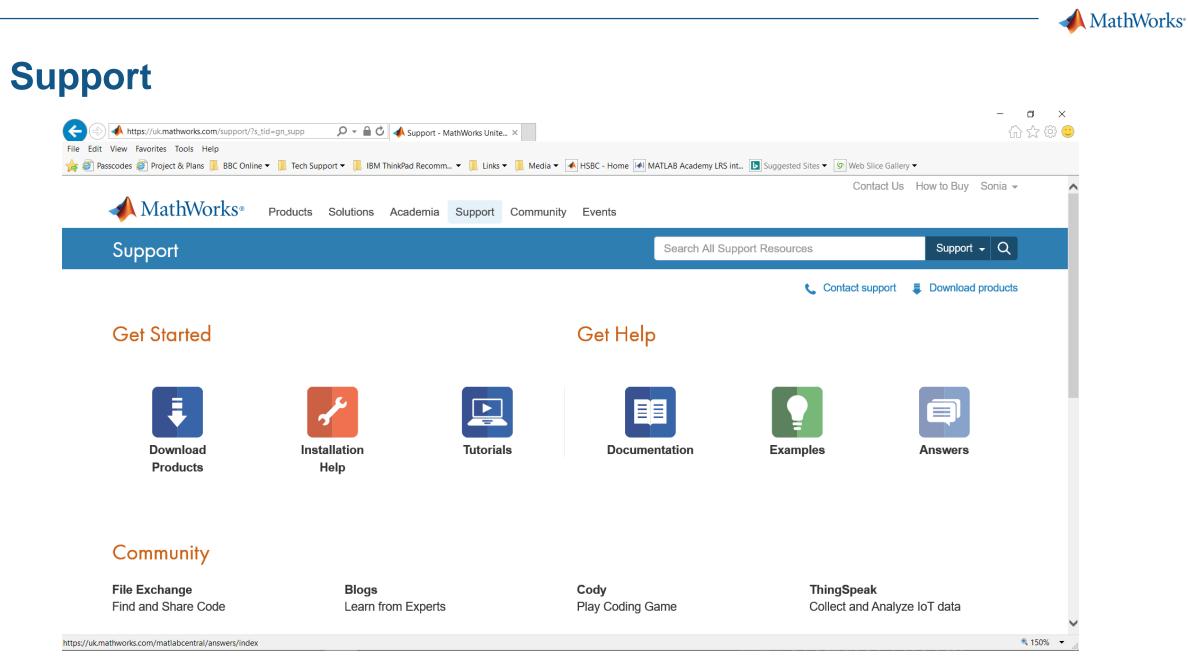
#### Summary

- Common challenges addressed!
  - Structured/ Common Environment
  - Graphical Dependency Analysis
  - Source Control Integration
  - Automation of common tasks
  - Options for sharing work
  - Parallel development workflows
  - Knowledge retention
- Simulink Projects for efficient team collaboration workflows
- Try it Today!



## **Additional Resources**

- Documentation
  - Project Management
- Example
  - Using a Simulink Project
- Tutorials
  - <u>Try Simulink Project Tools with the Airframe Project</u>
  - Create a New Project to Manage Existing Files
- Training
  - Simulink Model Management and Architecture
- Consulting
  - Proven Solutions from MathWorks Consulting Services





#### **Q & A**