MATLAB EXPO

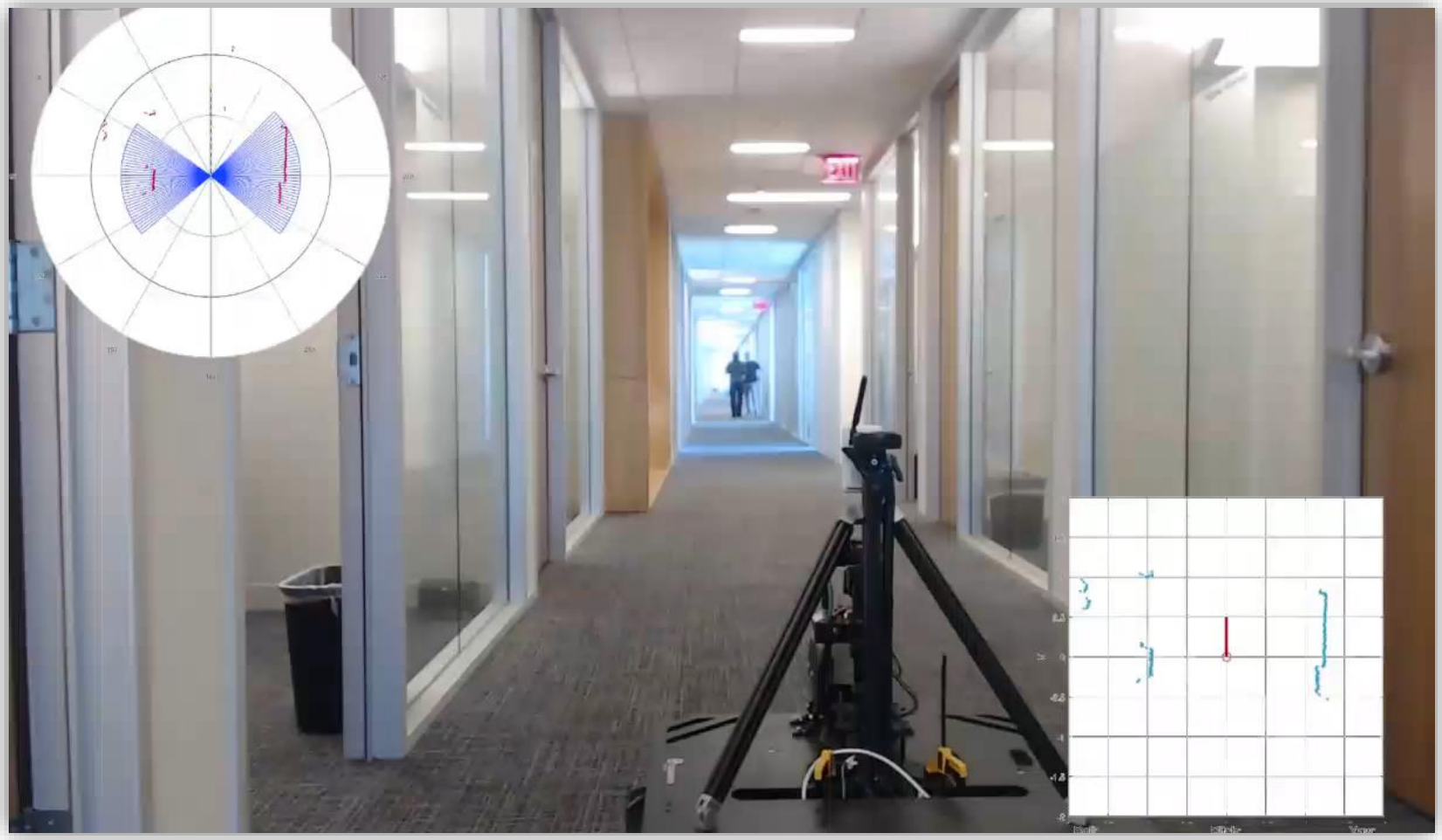
5 月 28 日 , 2024 | 北京

Simulink 通用集成接口助力系统级别仿真

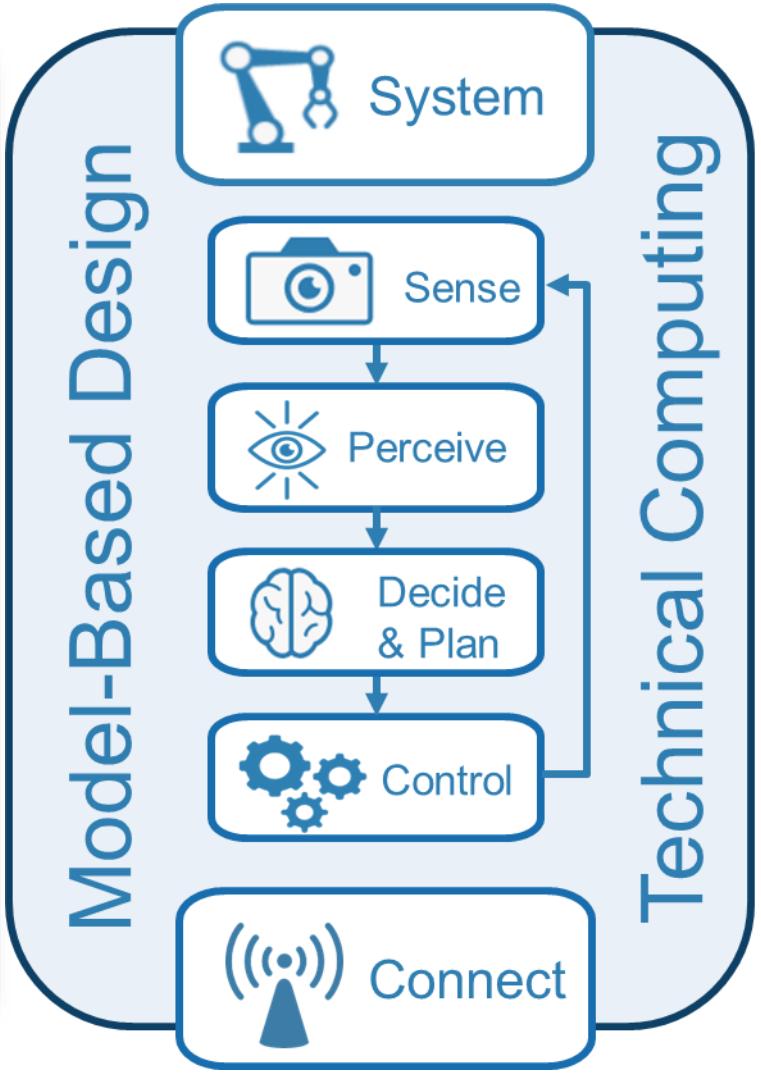
杨超, MathWorks

MathWorks

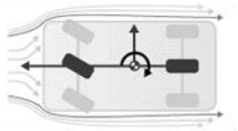
复杂系统涉及更多的学科/物理域



SIMULINK®

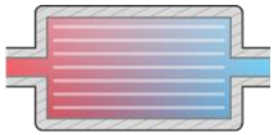


系统级别的仿真需要更加便捷的集成复杂接口

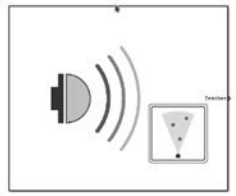


Simscape

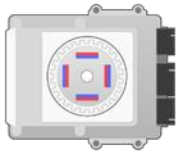
3rd party tool



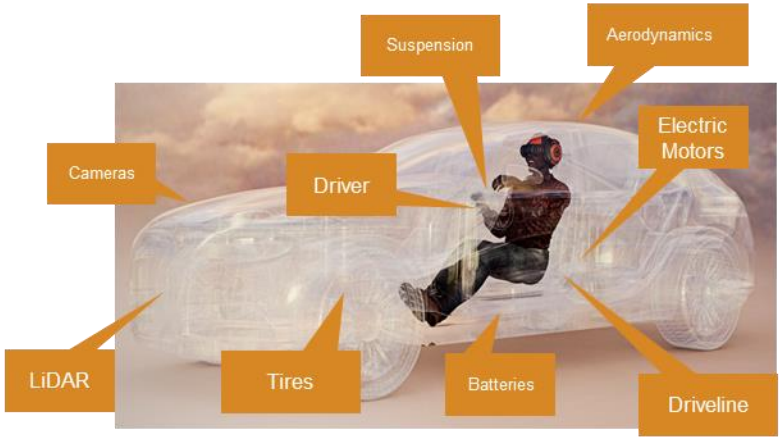
3rd party tool



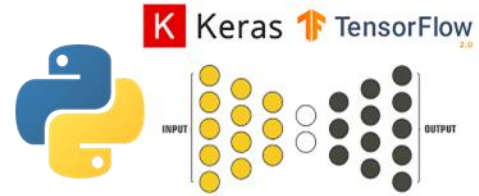
MATLAB



Windows

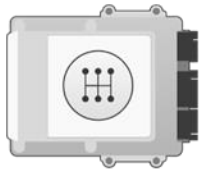


Linux



Python

fmi

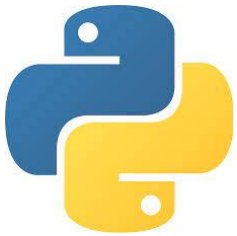


S-Function

C/C++ Code

多学科的系统级别仿真-多平台的便捷集成

自定义 C/C++ 集成



Python 集成

FMU 导入/导出

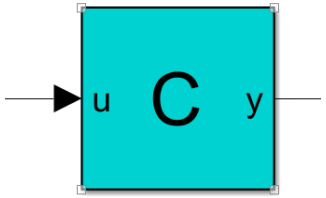
SIMULINK®

Simulation and Model-Based Design

自定义 C\C++ 集成

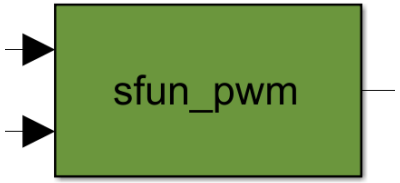
C Caller

- 简单地调用一个函数
- 自动创建 I/O 接口
- 导入共用库和 C 代码进行单元测试的理想选择



C Function

- 调用多个函数
- 支持在模块中实例化 C++ 类对象和调用类方法

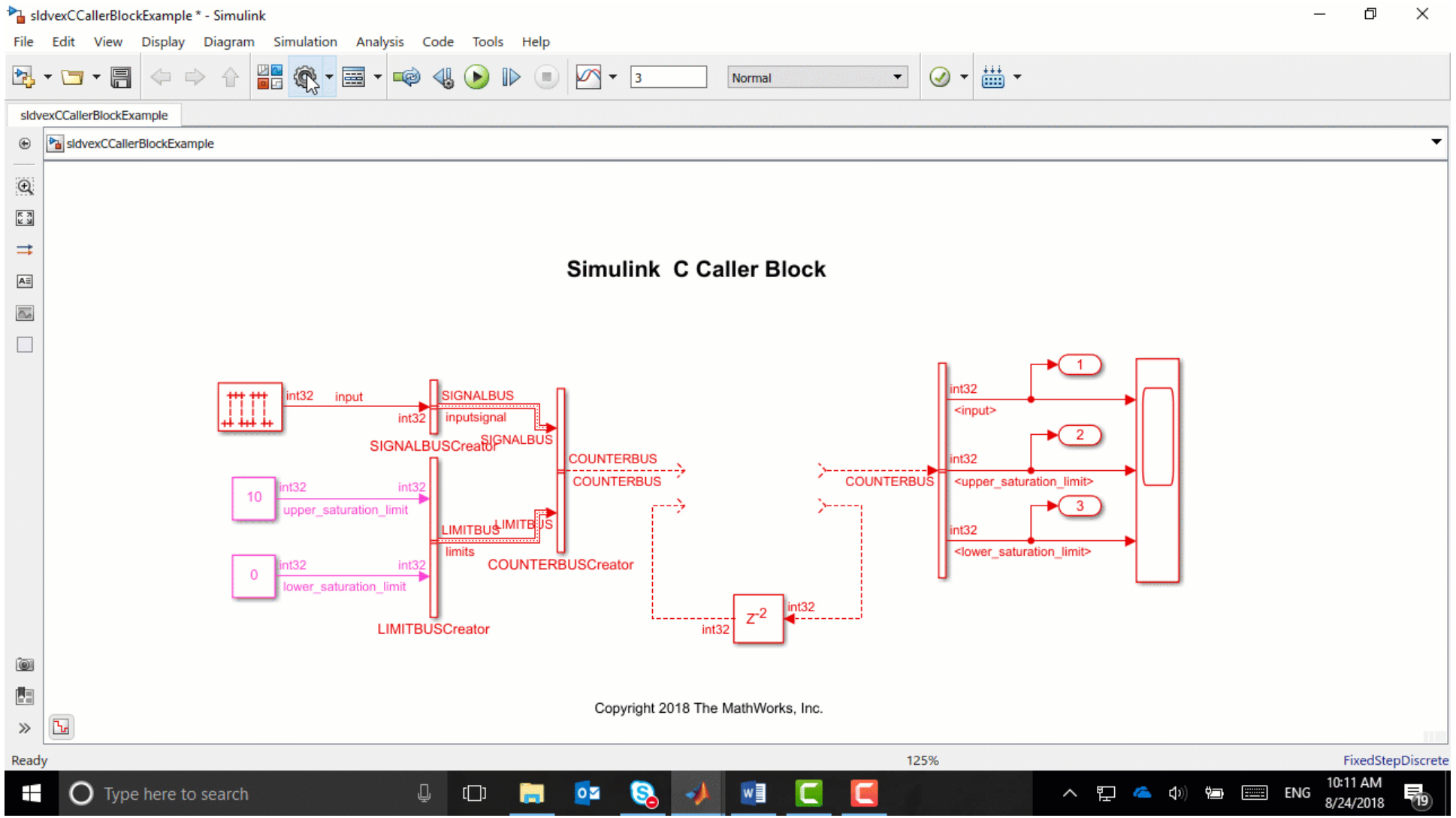


S-Function & S-Function Builder

- 通用性强
- 集成动态系统的理想选择

复杂度

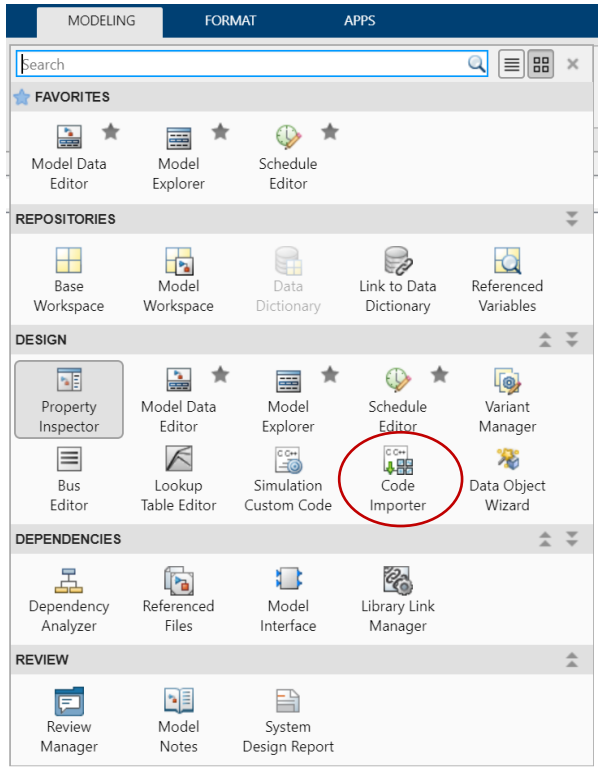
Demo 演示 – C Caller 模块集成单一 C 函数

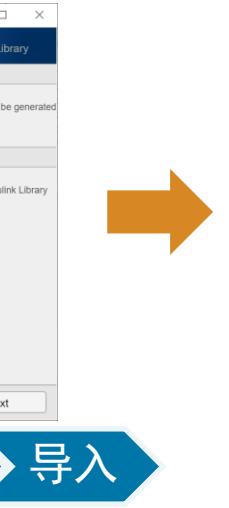


比 S 函数和 Legacy Code Tool 操作更简单

支持仿真和代码生成（集成）

Simulink 代码导入器-将自定义 C/C++ 代码转换为可复用的 Simulink 库





相关性 → 分析 → 匹配 → 导入

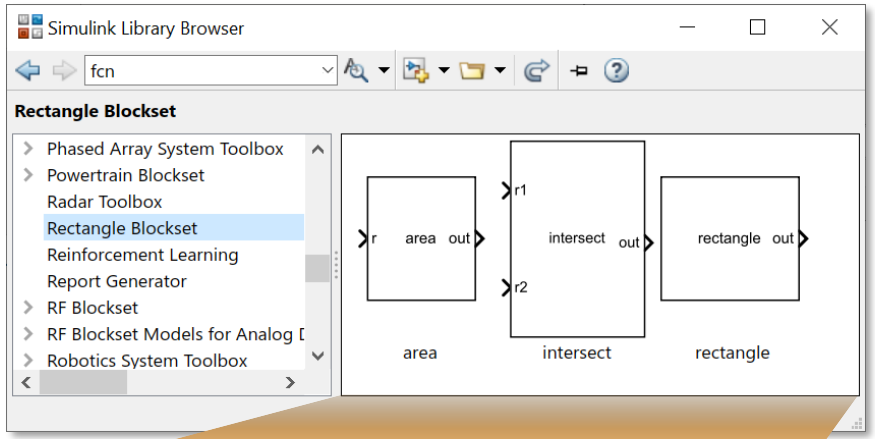
交互式向导提供了手把手的逐步指导

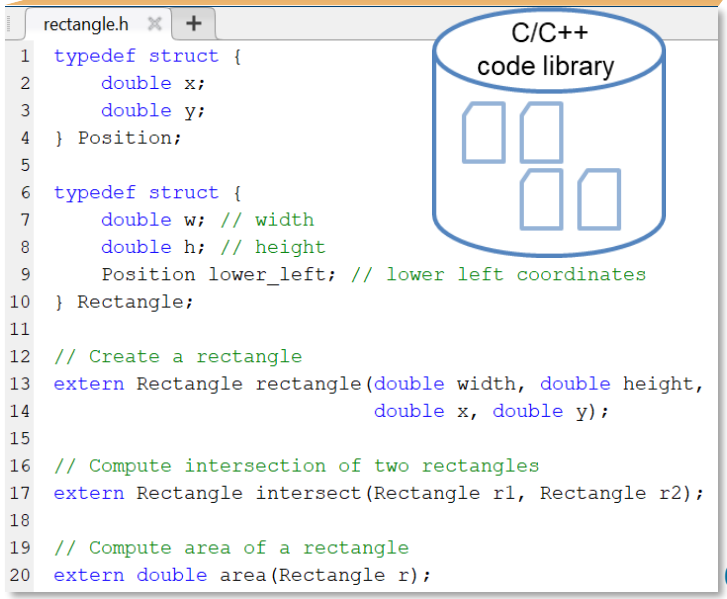
也提供 **MATLAB APIs** 的直观设置

https://www.mathworks.com/help/simulink/code-importer.html

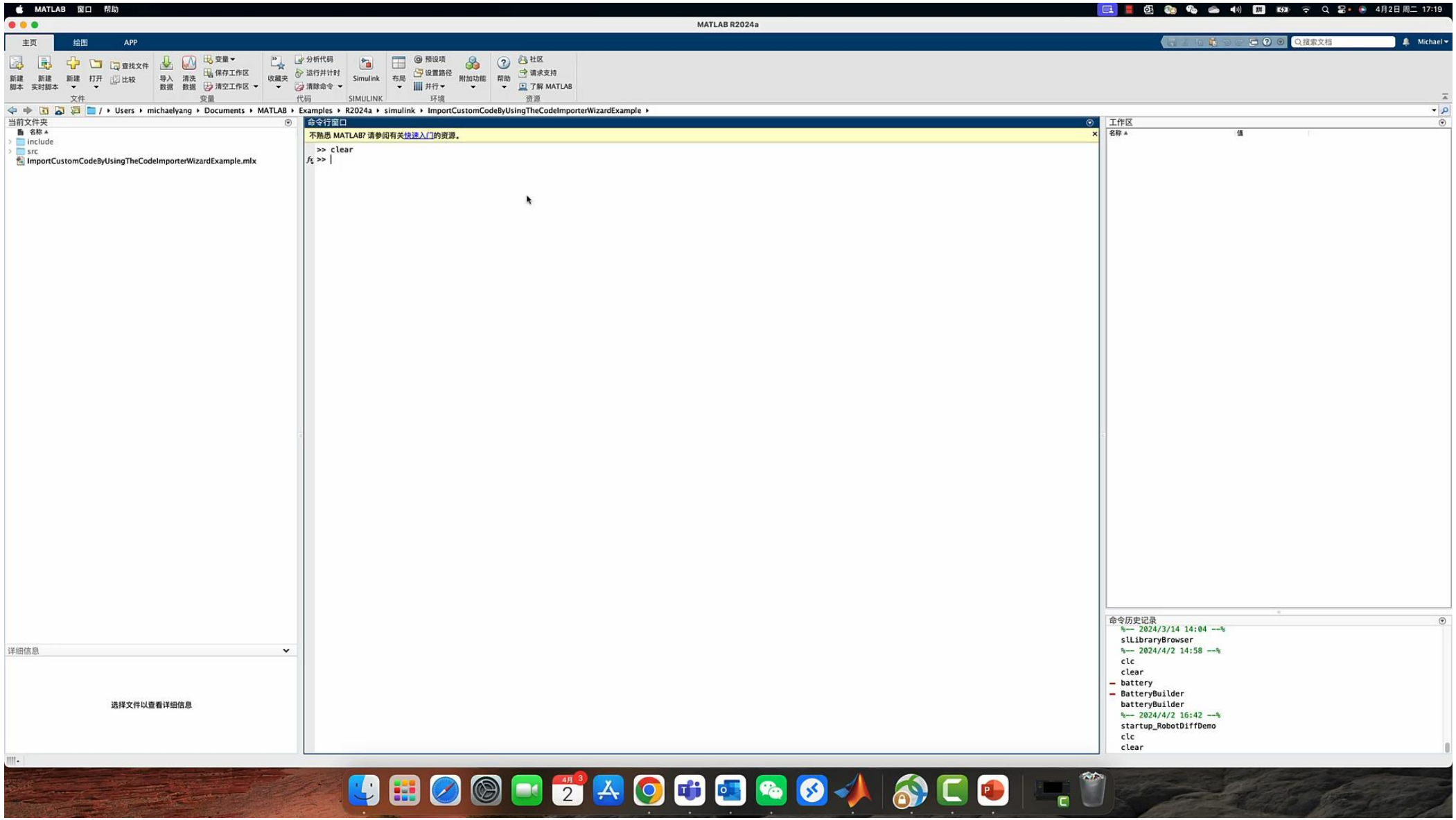
将 C 代码转换为**可复用的 Simulink 库**

使用 C caller 来代表 C 代码算法的模块





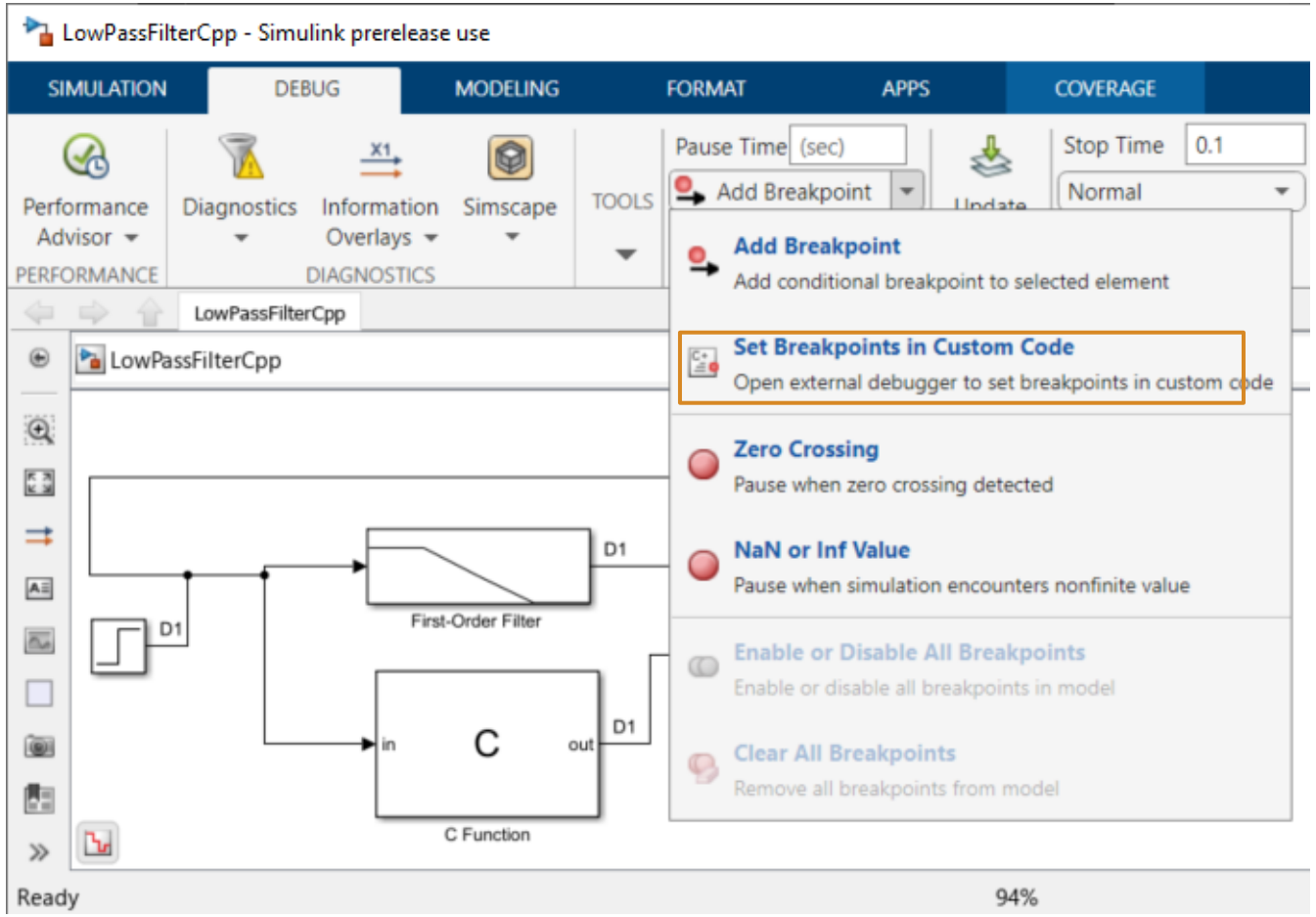
Demo 演示 - Simulink 代码导入器



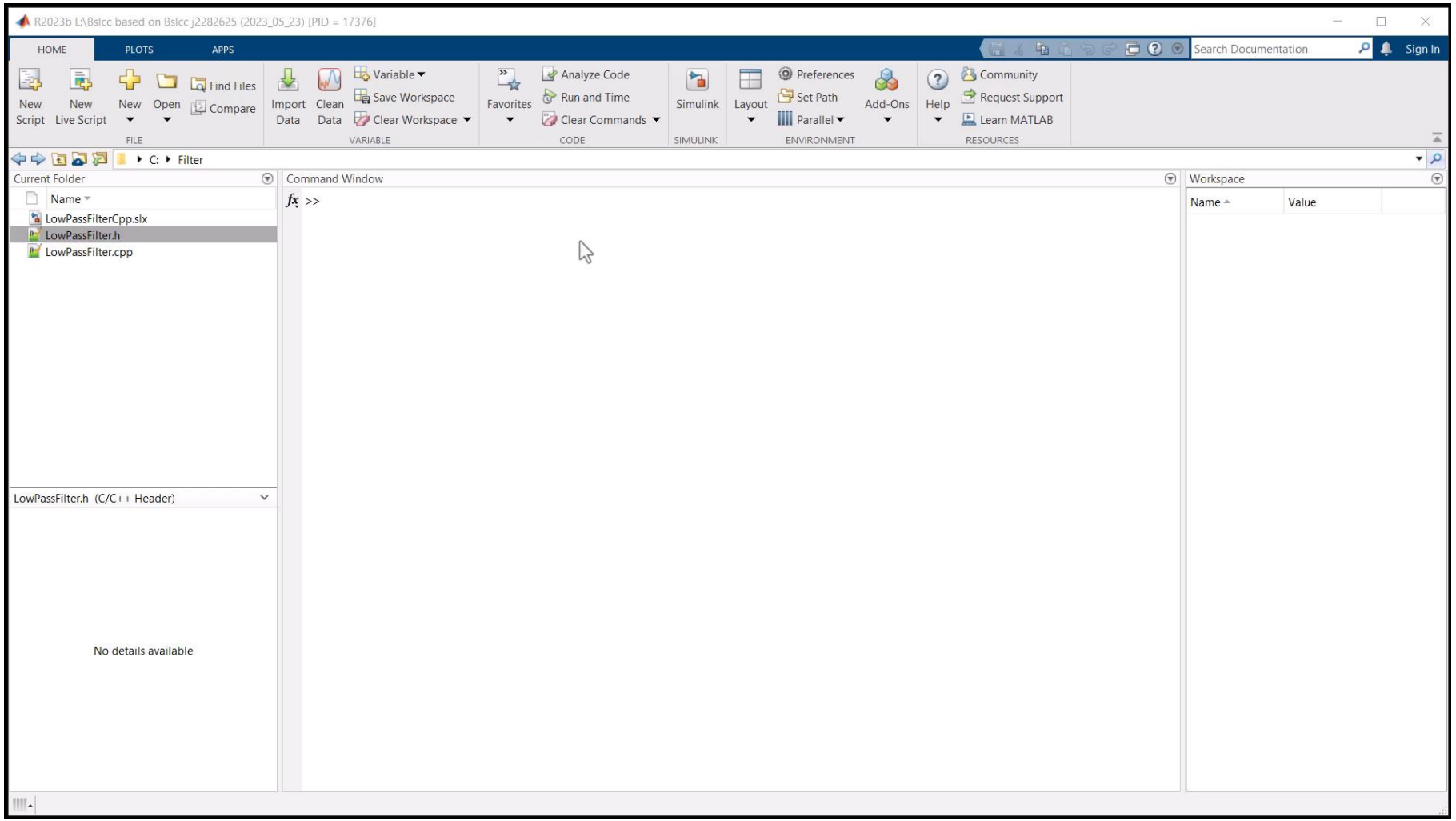
调试自定义代码

- 使用 Visual Studio 调试工具，将 MSVC 作为 MEX 的编译器

- Visual Studio code with
  - MINGW - gdb (Windows)
  - GCC - gdb (Linux)
  - Clang - LLDB(Mac)

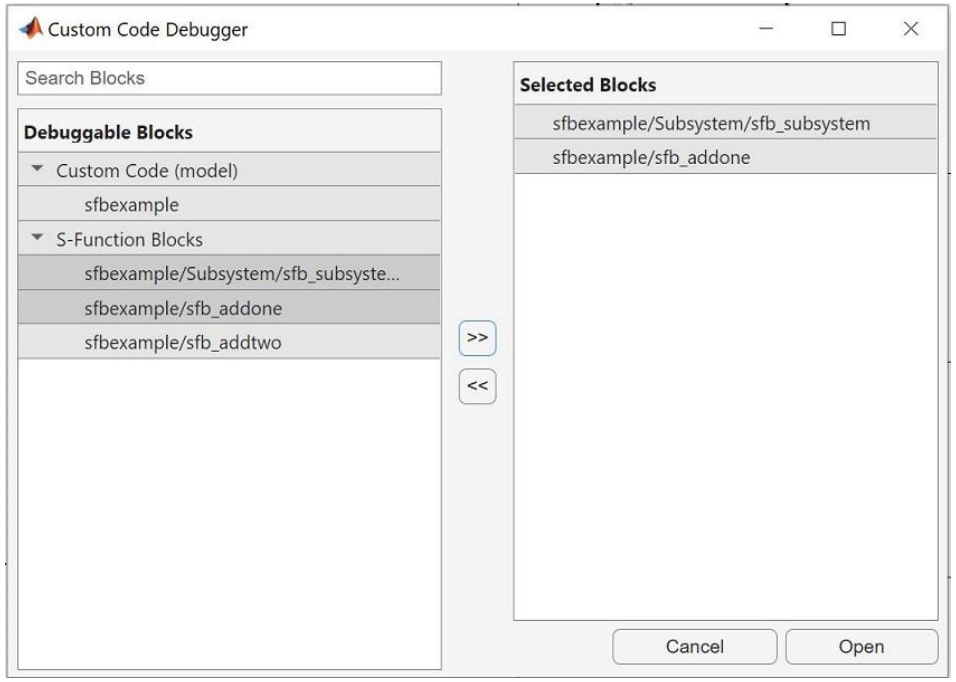


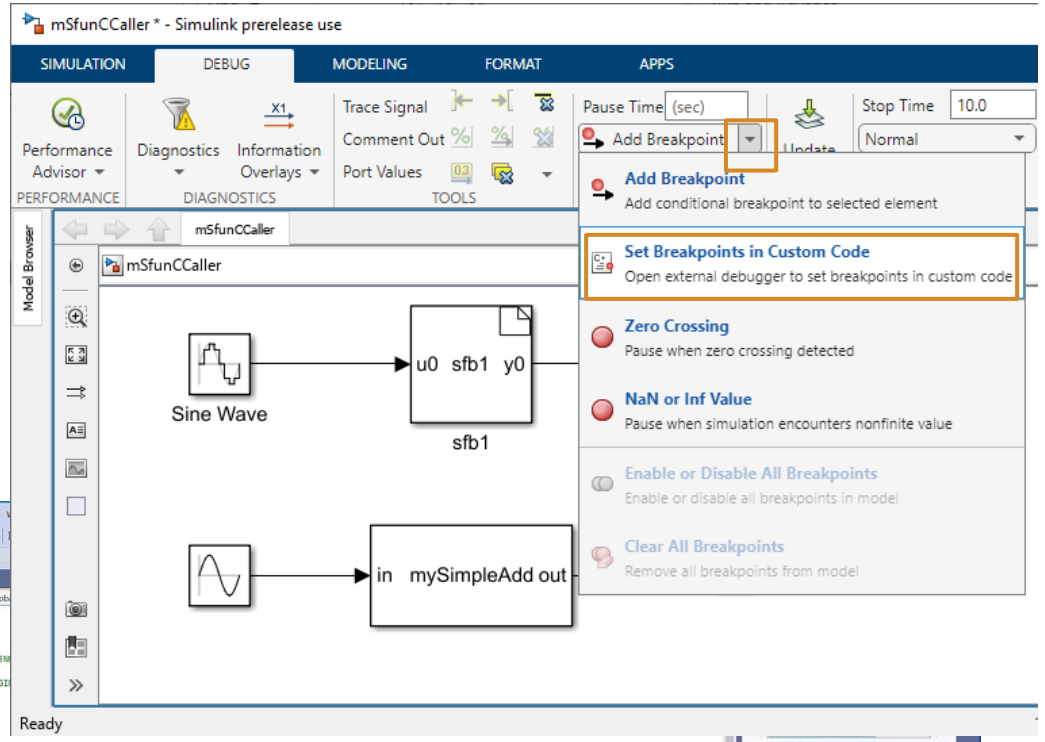
Demo 演示 – C++ 类支持与调试

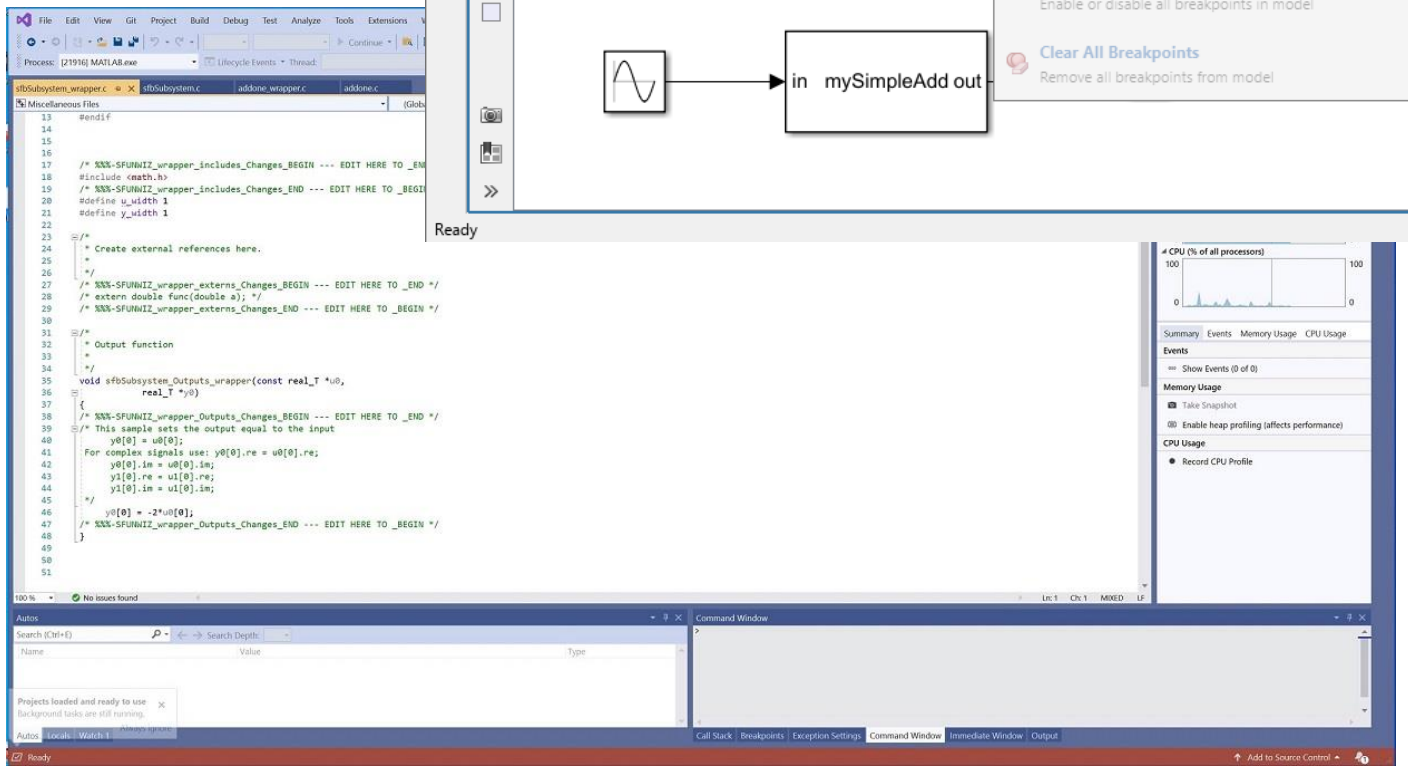


调试自定义代码

- 为选定的 S-函数自动运行调试器
- 设定断点并调试 C/C++代码

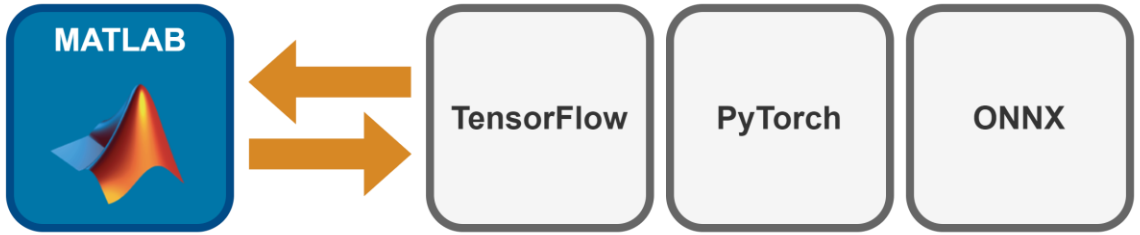




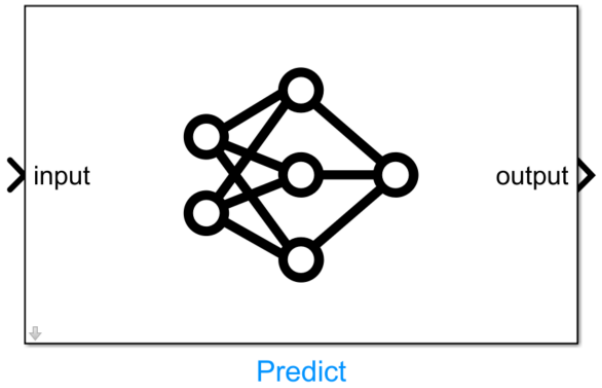


Python 集成 – 从 Python 导入深度学习模型

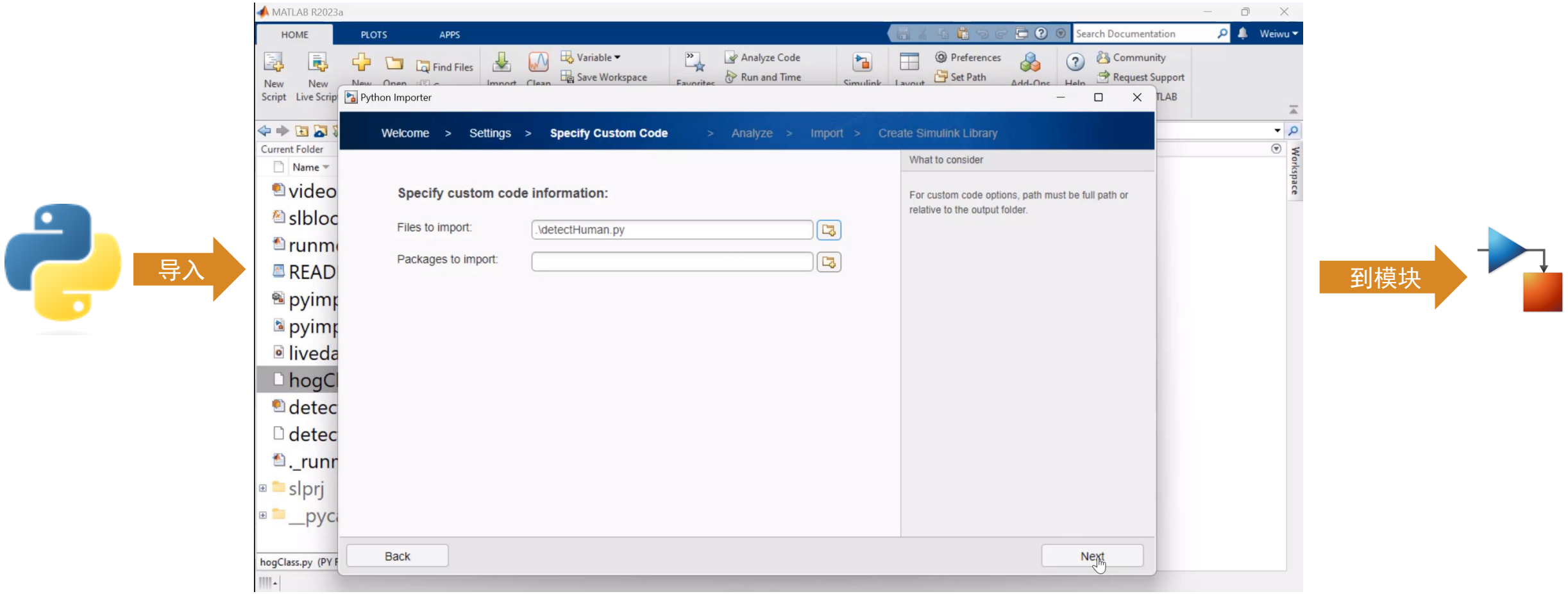
将 TensorFlow, PyTorch, 和 ONNX 模型导入到 MATLAB 代码



- 一旦模型转换到 MATLAB 中，便可以将其通过深度神经网络模块放到 Simulink 中进行仿真

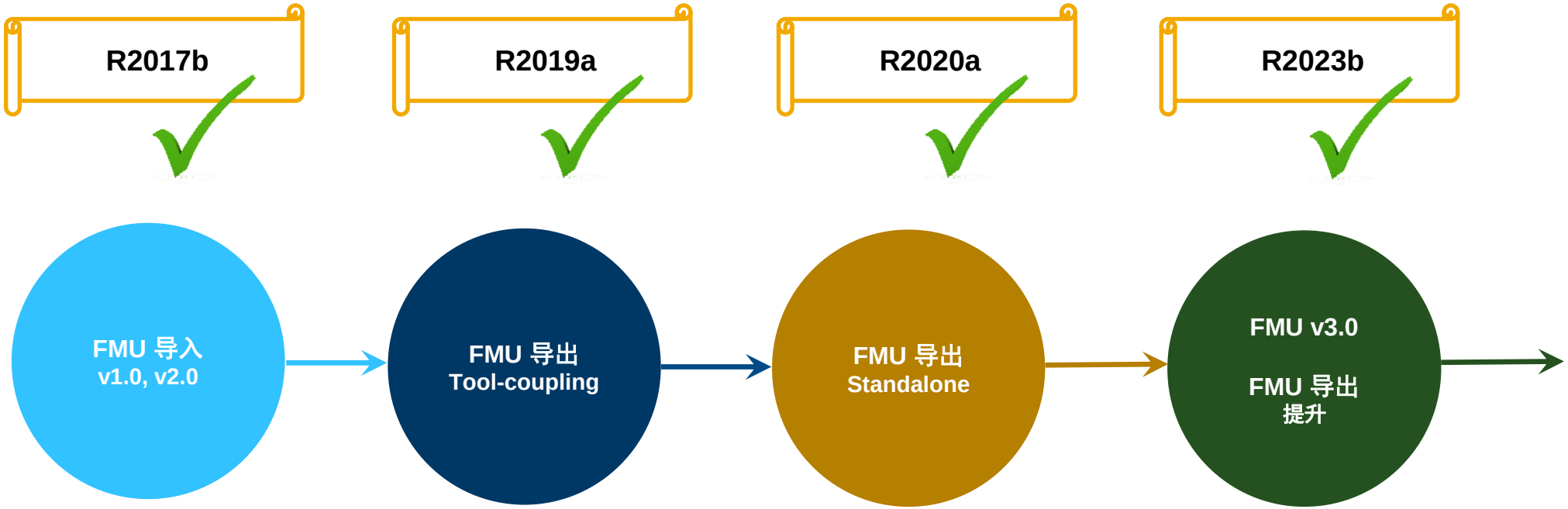


Python 集成 - 使用 Python 导入器从 Simulink 调用 Python 代码



- 图形化向导提供了逐步的操作指导，最小化手工代码编写
- 将单个 Python 函数组成的 Python 函数包集成到一个相关的库模块中
- 方便复用和构建自定义工具集

FMI 集成 – 路线图



阶段 I:

FMU 1.0, 2.0 导入:

FMU 模块 (Co-Simulation and Model Exchange)

阶段 II:

FMU 2.0 导出:

Tool-coupling Co-Simulation

阶段 III:

FMU 2.0 导出:

Standalone Co-Simulation (定步长求解器)

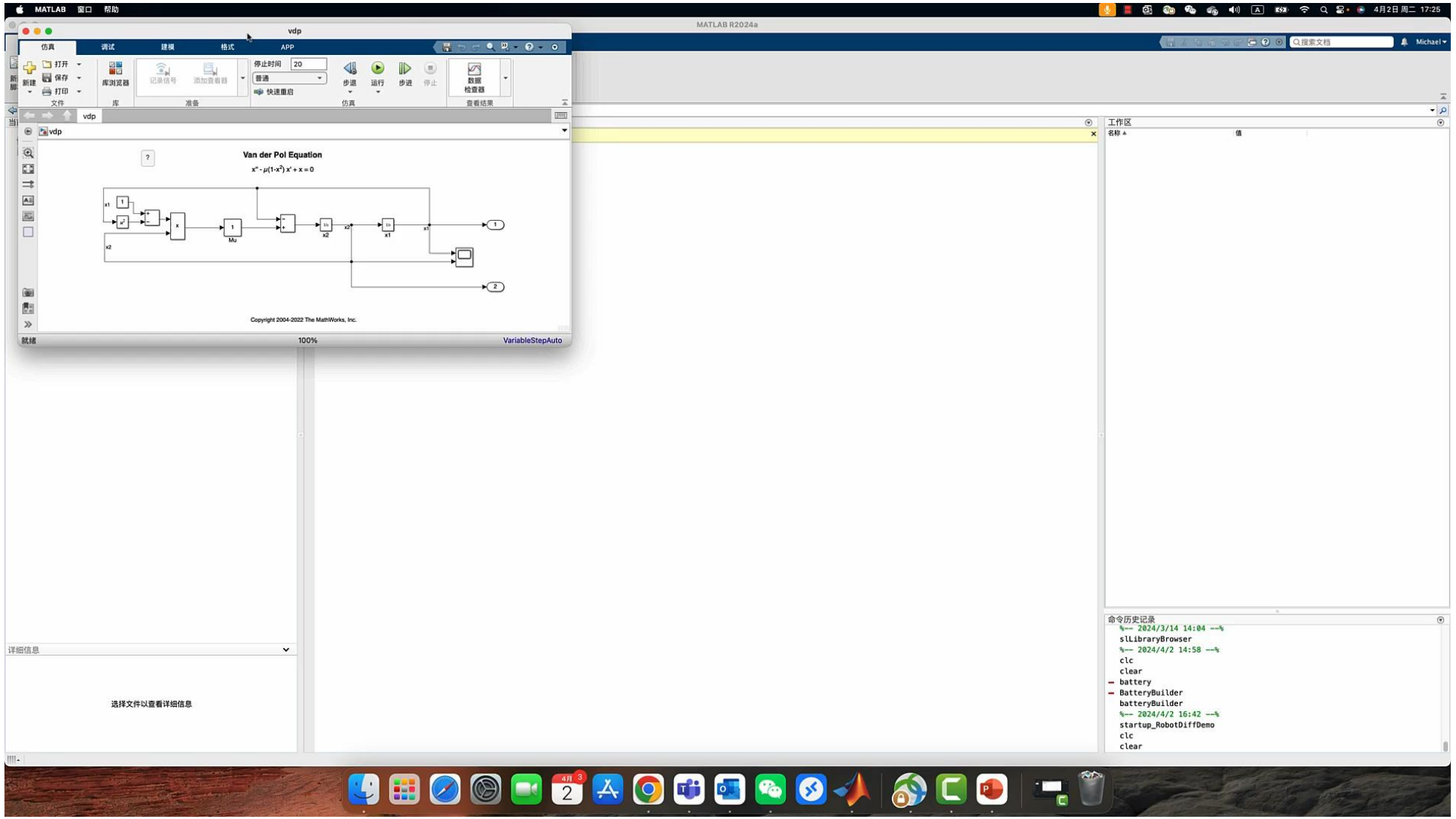
Phase IV:

FMU 3.0 导入
(model exchange, co-simulation)

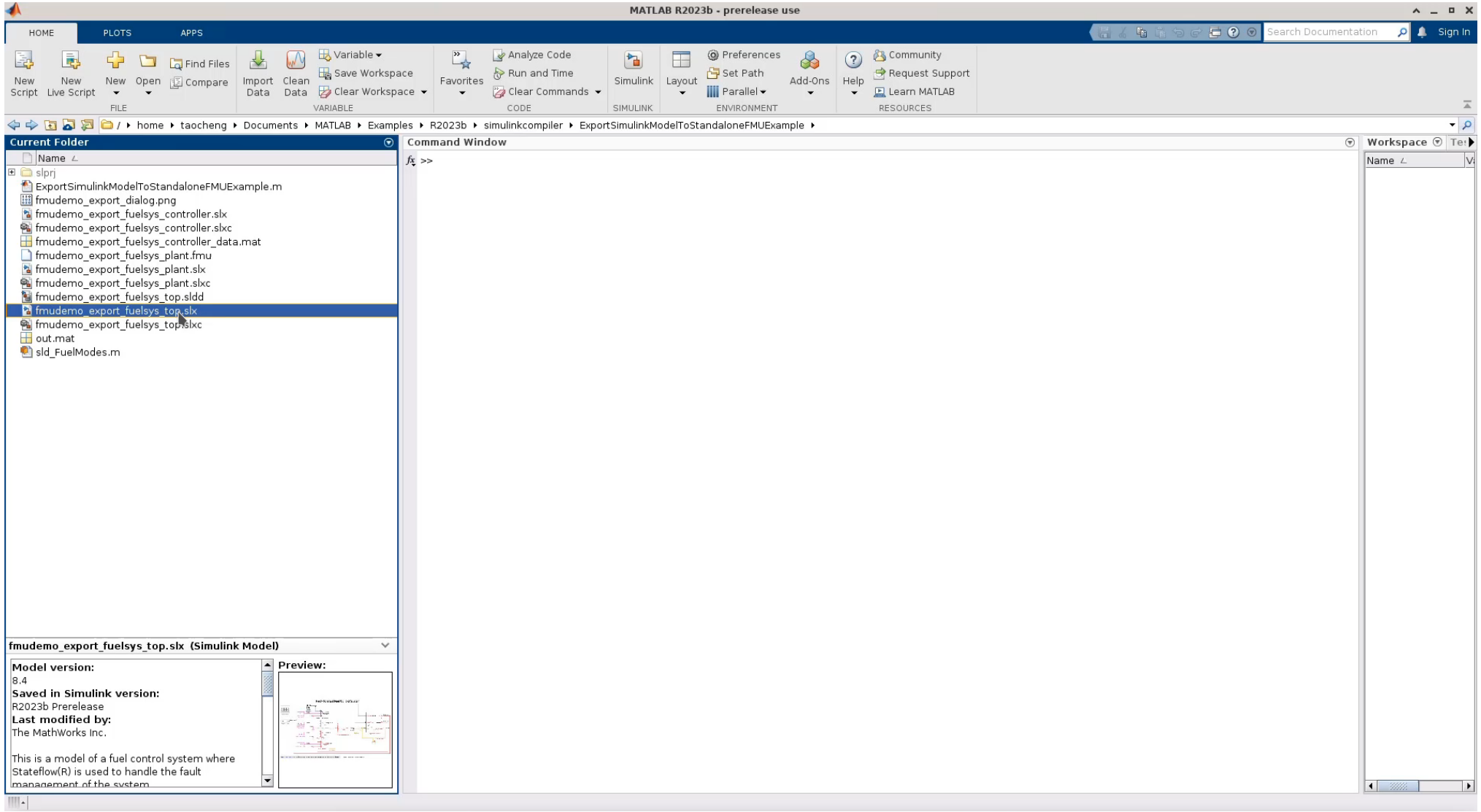
FMU 3.0 导出
(定步长求解器 , co-simulation)

FMU 2.0 导出
(变步长求解器 , co-simulation)

Demo演示 – 从 Simulink 导出 Tool-coupling FMU



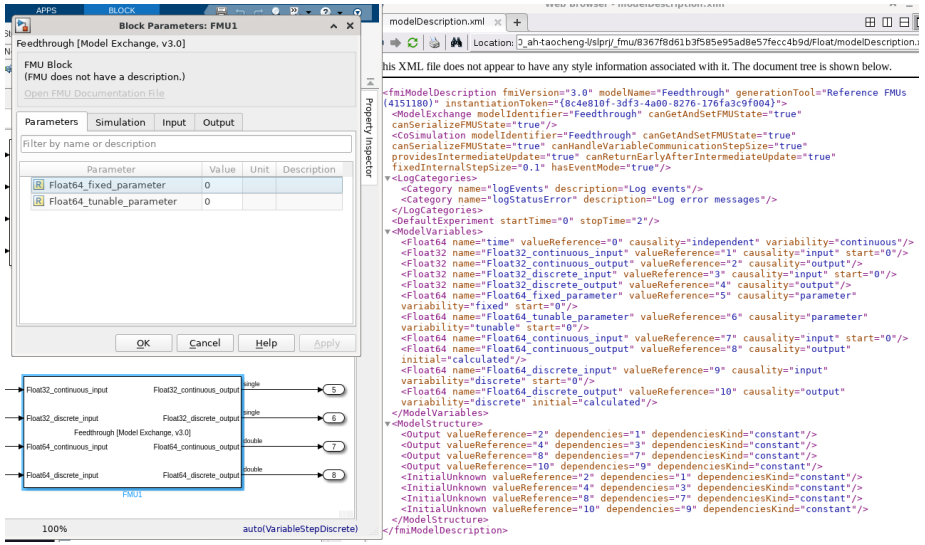
Demo演示 – 从 Simulink 导出 Standalone FMU



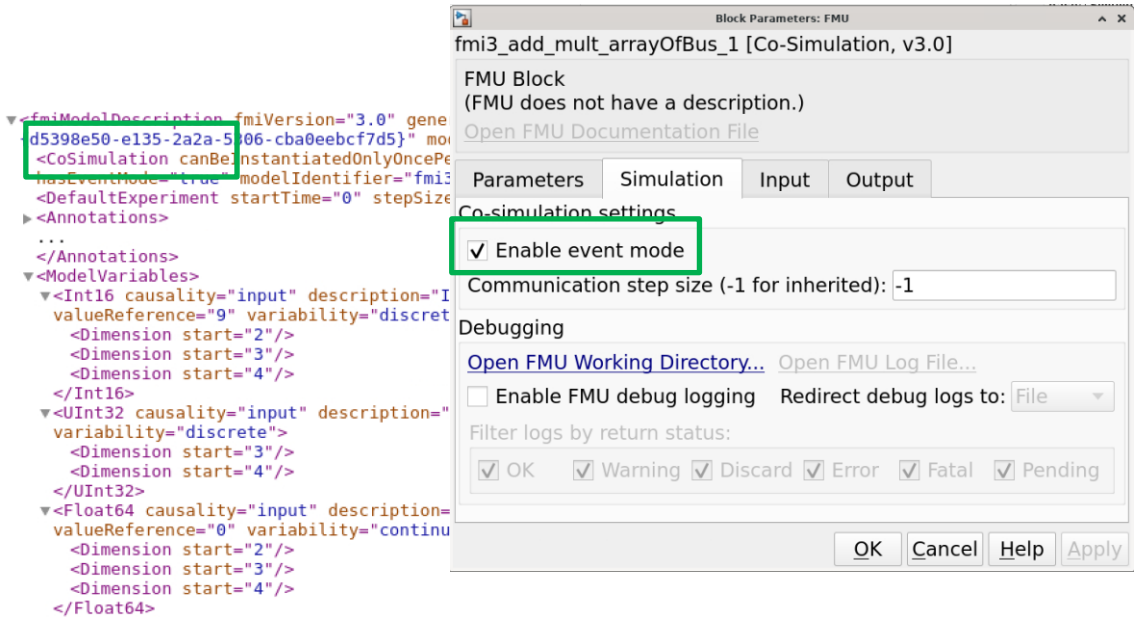
FMI 3.0 支持: FMU 导入模块

- FMI 3.0 (2022年5月发布)
  - https://fmi-standard.org/docs/3.0-dev/

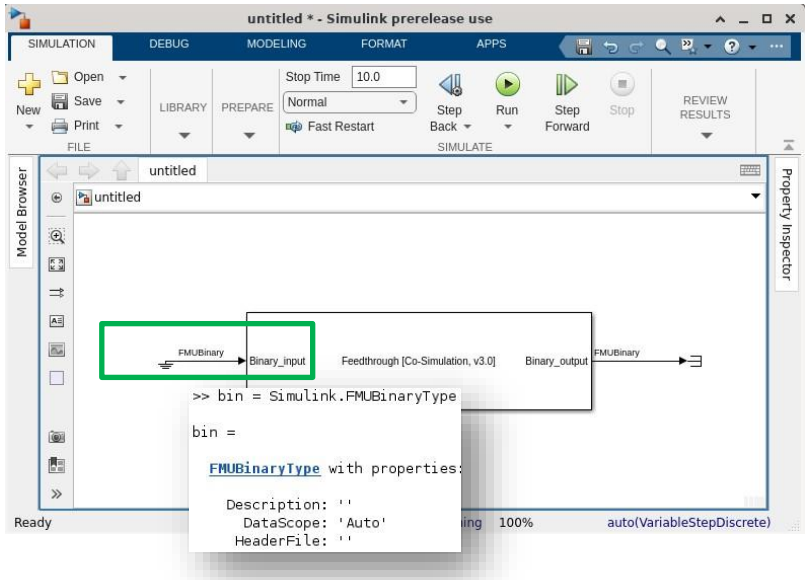
- Simulink 从 R2023b 开始支持 FMI 3.0 导入



FMU Import block loading FMU 3.0 modelDescription file



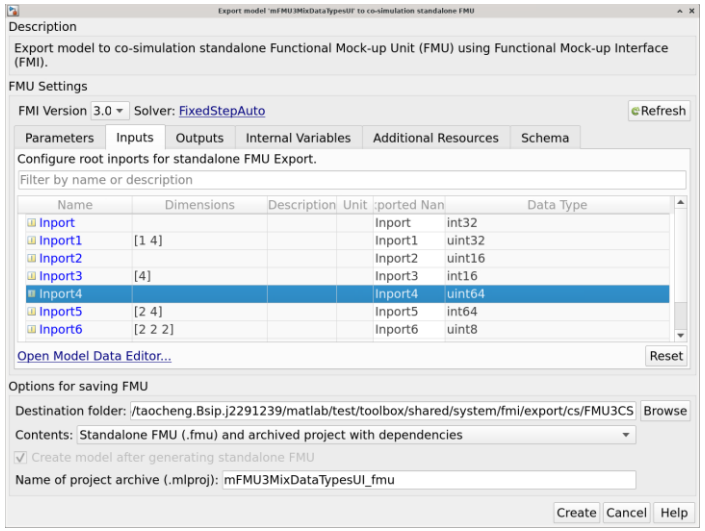
FMU Co-simulation with event mode eliminates one-step delay



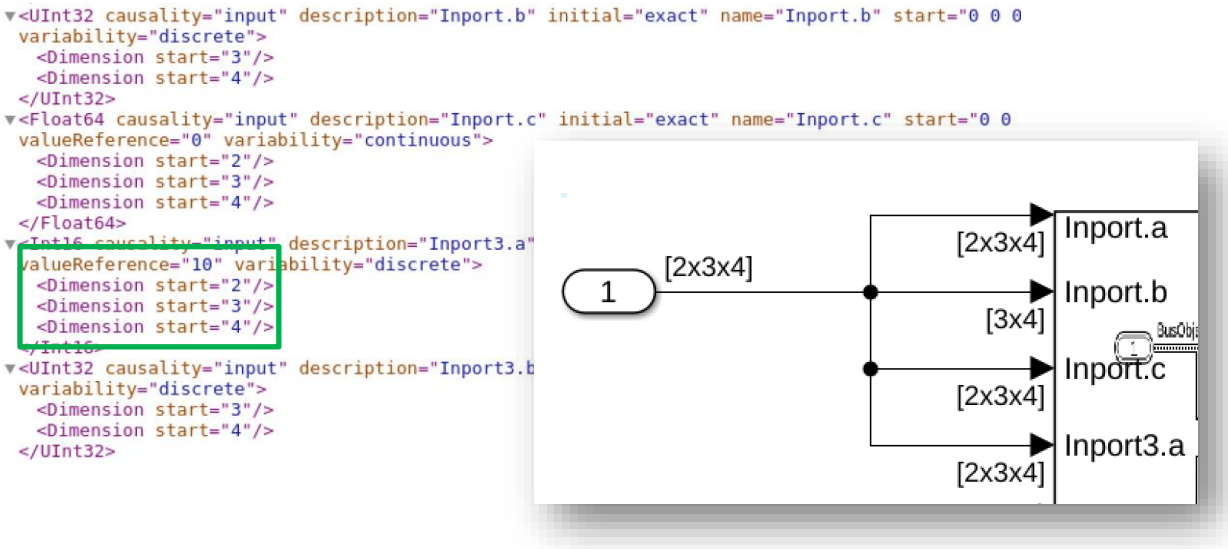
FMU Binary data type

FMI 3.0 支持: Standalone FMU 导出

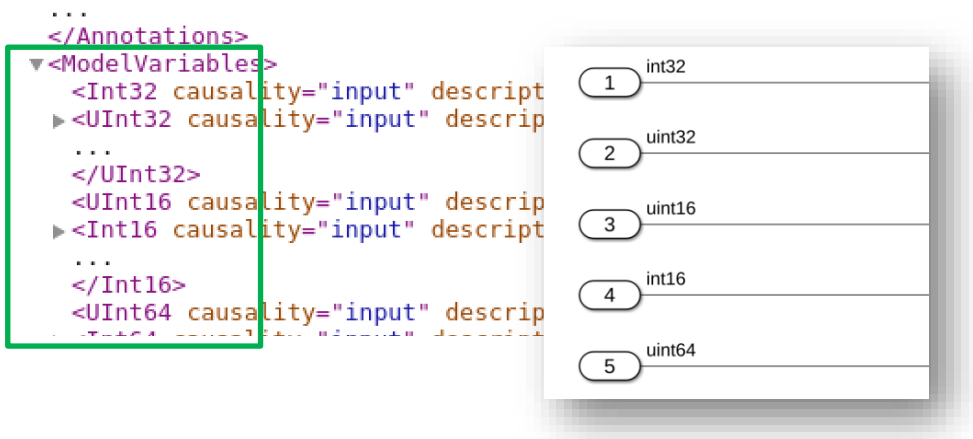
- Simulink Compiler 支持在 R2023b 中导出 FMU 3.0
  - 通过 **Simulink FMU Builder** Support Package
  - FMU 模式：Co-Simulation mode



FMU Standalone Export dialog

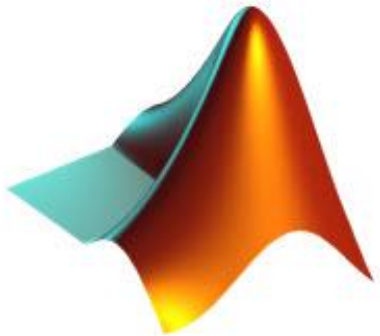


Export Array/Matrix signals to FMU 3.0



Export new int datatypes to FMU 3.0

多学科的系统级别仿真-多平台的便捷集成



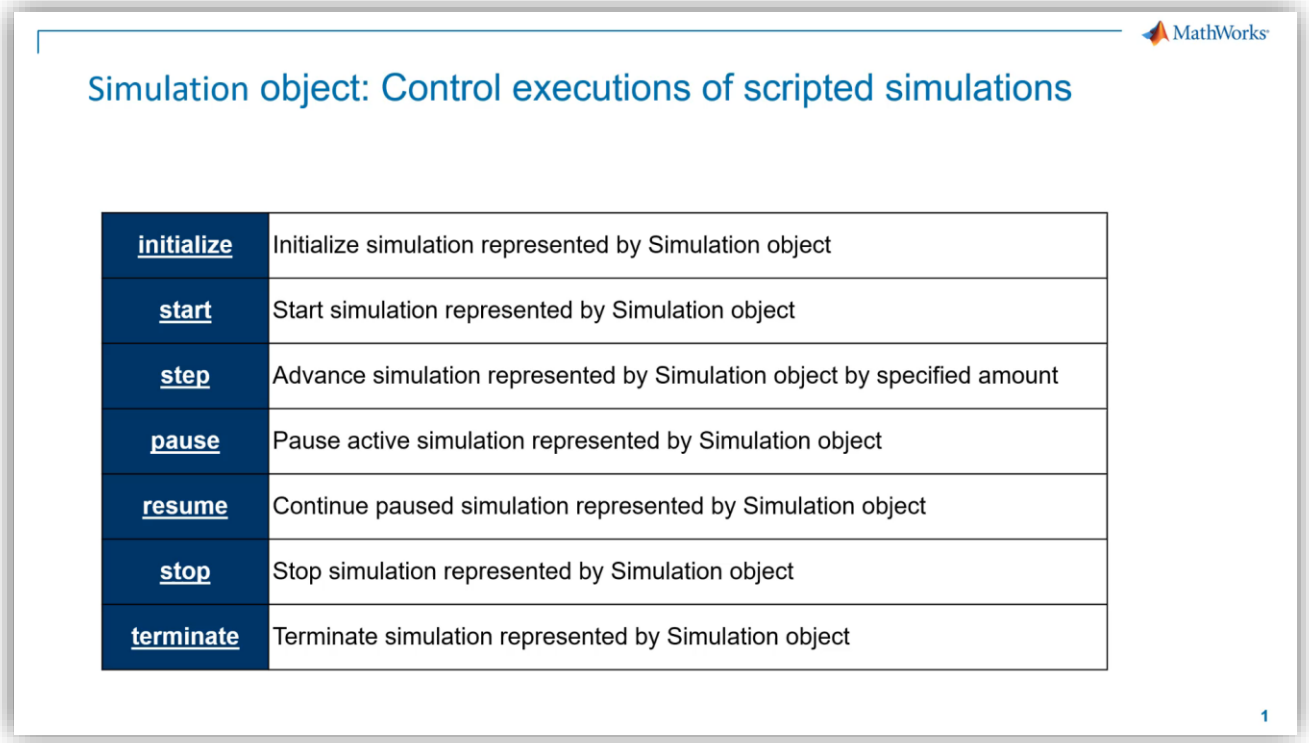
R2023b R2022b
R2023a
R2024a

SIMULINK®
Simulation and Model-Based Design

仿真对象

使用仿真对象（Simulation object）来控制基于脚本的交互式仿真

- 通过脚本程序来用更细的颗粒度控制仿真执行：包括初始化、启动、步进、暂停、中止等等
- 在仿真执行时通过脚本程序调节仿真参数
- 支持所有的仿真模式、包括 rapid accelerator，以及通过 Simulink Compiler 实现部署的场景

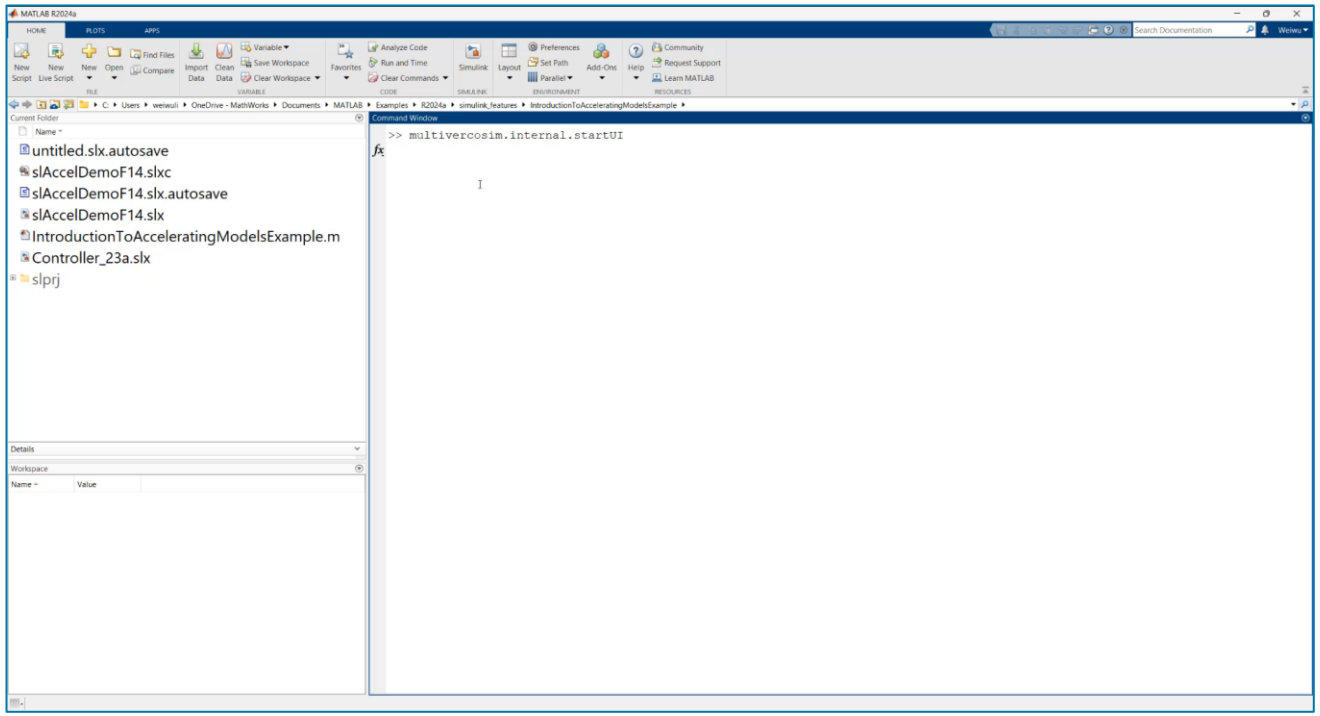


多版本联合仿真支持包

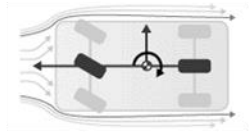
将不同版本的 Simulink 模型集成在一起以用于联合仿真

- 使用多个 MATLAB 会话实现不同版本模型的联合仿真
- 隔离各自的工作区、路径、求解器和版本
- 支持做调试
- 支持无法生成代码的模型

- *对应版本的 MATLAB 需要安装在本机上*

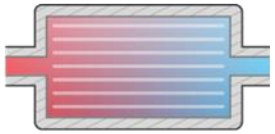


总结-Simulink 能够提供系统级别的仿真需要的复杂集成接口

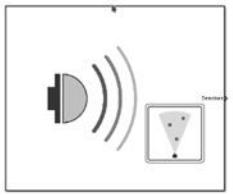


Simscape

3rd party tool



3rd party tool

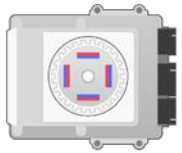


MATLAB

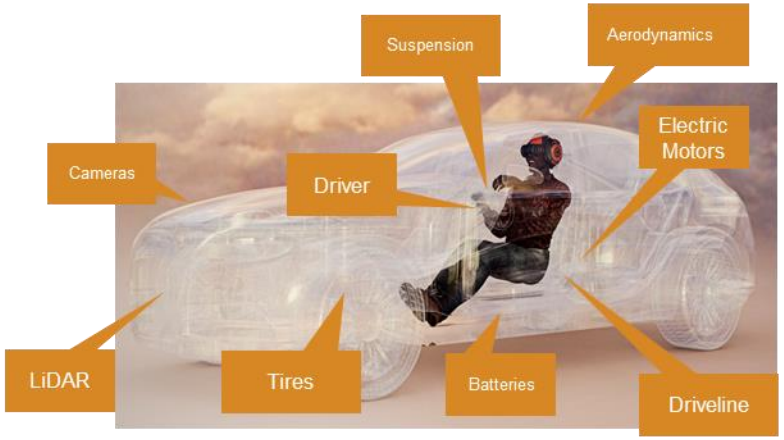
使用仿真对象（Simulation object）来控制基于脚本的交互式仿真

将不同版本的 Simulink 模型集成在一起以用于联合仿真

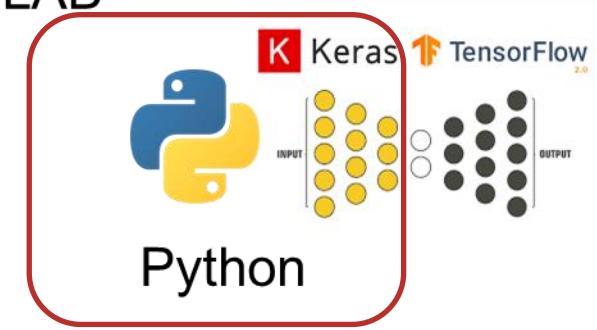
SIMULINK®

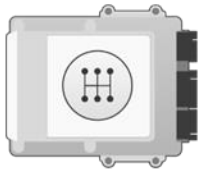


Windows

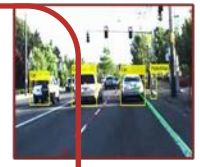


Linux





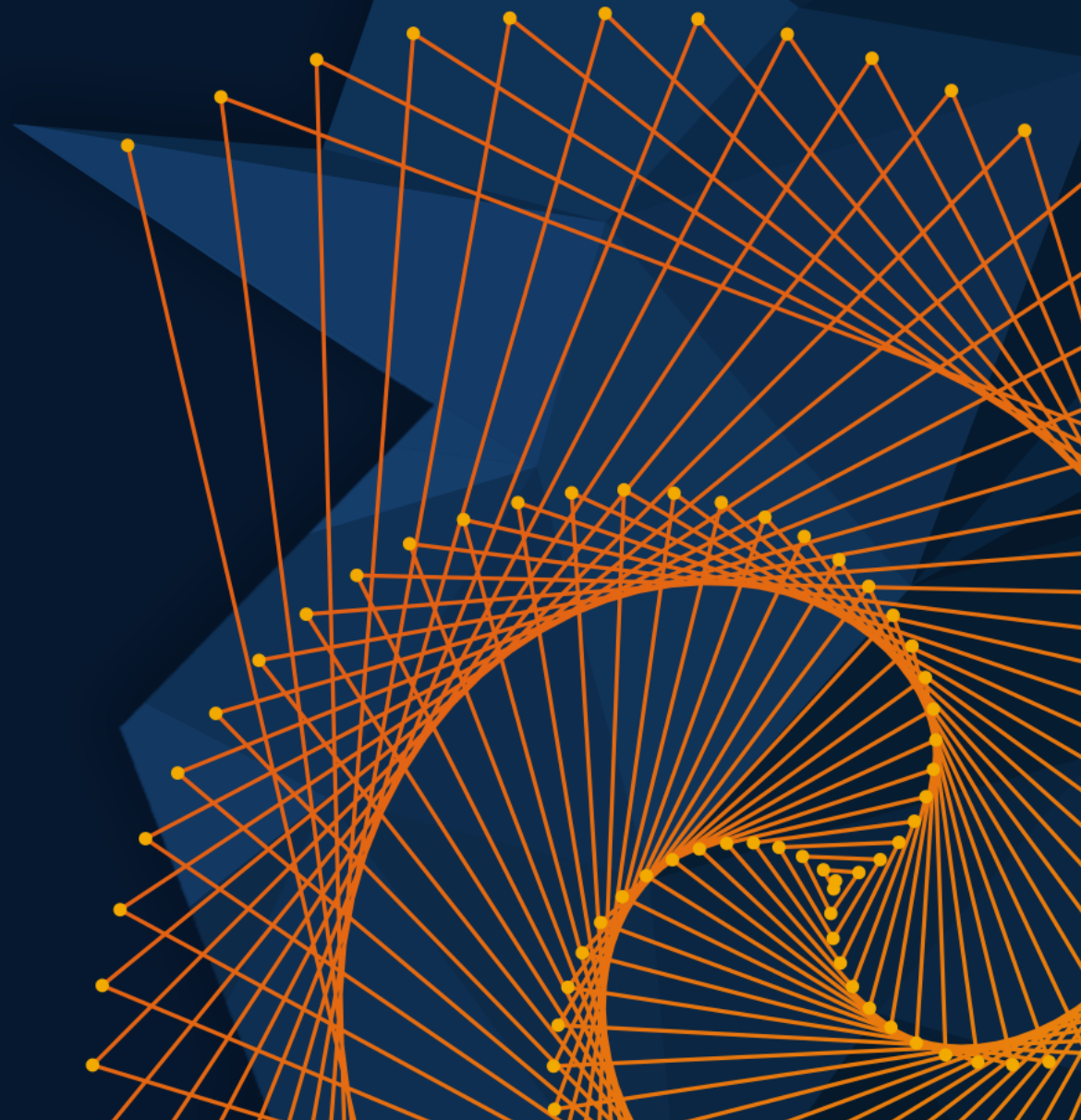
S-Function



C/C++ Code

MATLAB EXPO

谢 谢



MathWorks®