



使用 **Simulink** 设计 面向服务的架构 (SOA)



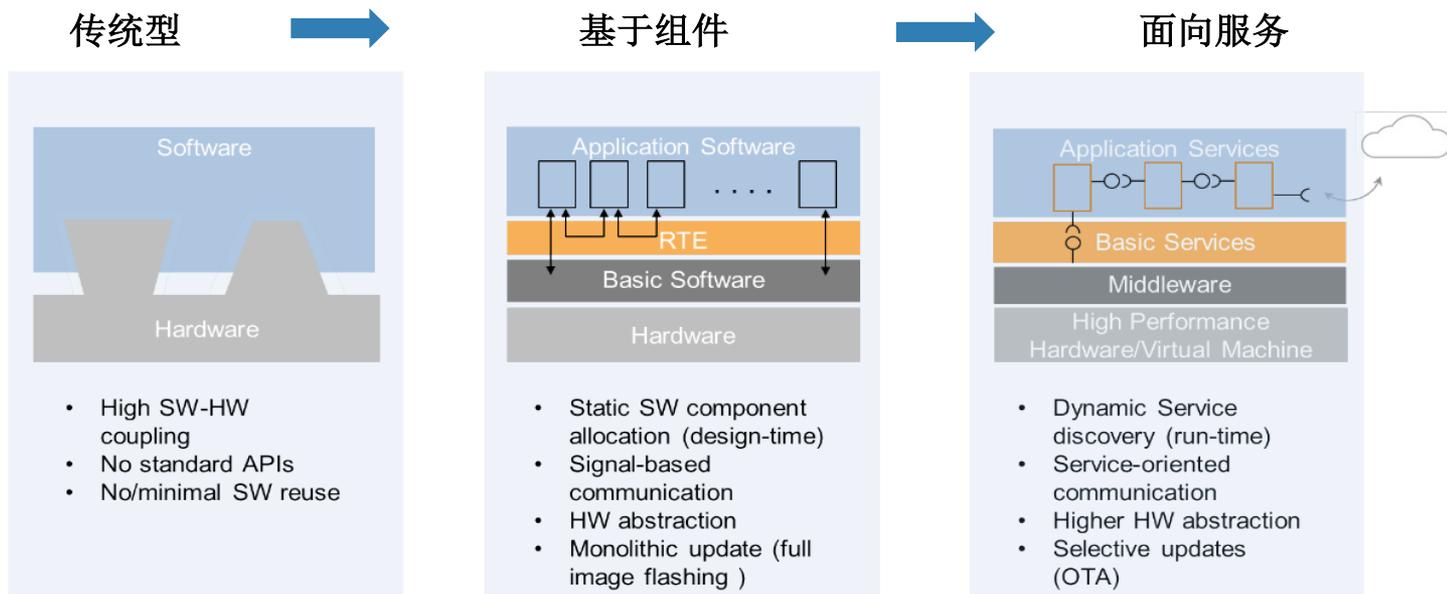
面向服务的架构概述

面向服务的架构 (SOA) 是一种软件架构，它描述了由一组通过消息通信的服务组成的系统。它通常涉及跨平台使用这些服务的多个应用程序。

SOA 支持灵活地动态添加、删除或更新组件，而不会影响整个软件系统。

本电子书描述如何使用 **Simulink** 对基于不同应用程序的 SOA 软件进行建模、仿真和部署。

面向服务的架构概述（续）



为了满足智能应用（例如自动驾驶）的需求，现代软件架构正在从传统架构演变为基于组件的架构，再到面向服务的架构。

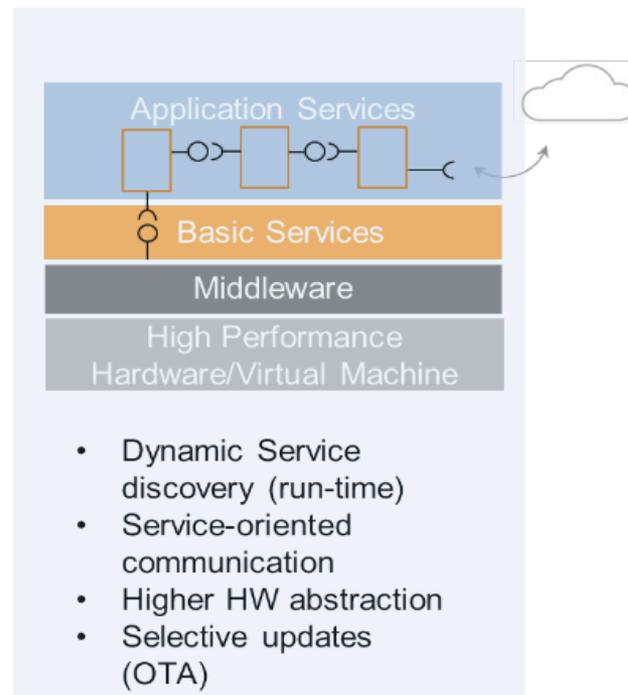
- 传统型：没有标准 **API** 的高度耦合的软件和硬件
- 基于组件：更加标准化，应用软件由静态分配给 **ECU** 的组件构成
- 面向服务：通过组合系统中可用的服务来开发应用程序

面向服务的架构概述（续）

SOA 提供将应用程序作为独立服务进行集成的架构框架，这些独立服务可以通过无线方式轻松地组合、更新或升级。

- 基于 SOA 的系统为一个包含多个应用程序和服务的组合物，这些应用程序和服务可以分布在高性能计算机上和云端。
- 这些系统使用面向服务的通信，而不是通过网络广播信号。
- 信息在服务提供方和订阅这些服务的使用方之间交换（优化带宽使用）。

面向服务



面向服务的架构的行业标准

AUTOSAR Adaptive 平台: AUTOSAR 组织基于 SOA 开发了此平台。该 Adaptive 平台在自动驾驶应用所需的处理分发和计算资源分配方面提供了灵活性和可扩展性。这样的话,即使在 adaptive ECU 软件发布后,也可以安全地更新和升级。

DDS: 数据分发服务 (DDS) 运用 SOA 方法学,直接处理实时和嵌入式系统的发布和订阅通信。DDS 用于航空航天、国防、汽车、机器人等行业的实时数据交换。

ROS: 许多机器人应用程序使用基于 SOA 方法的机器人操作系统 (ROS)。ROS 充当运行软件所必需的组件相互通信的框架。

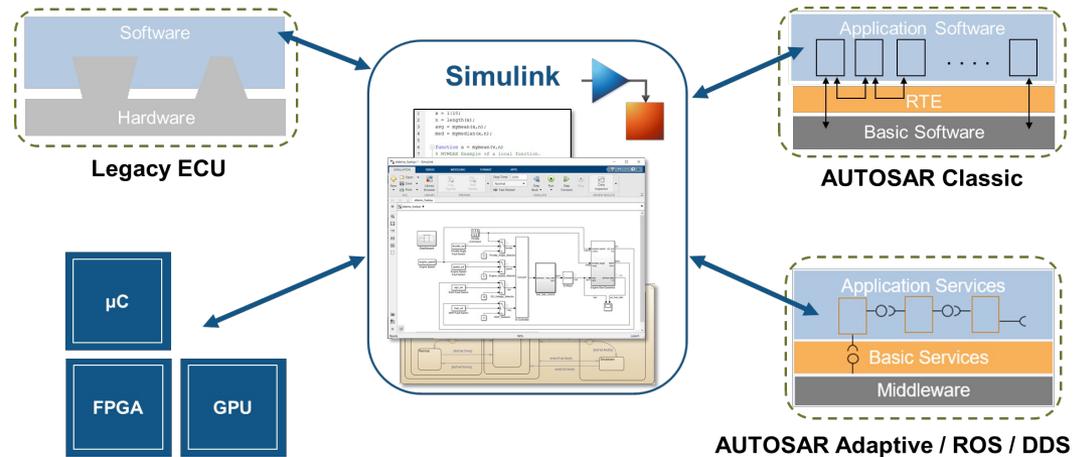


Simulink 支持面向服务的架构

工程师可以重用已建立的工作流，并将其经过测试的现有组件迁移到新架构。

使用 MATLAB 和 Simulink 附加组件，您可以只进行一次软件开发，并将其部署到多个目标。

您可以将您的模型部署到传统 ECU、AUTOSAR Adaptive/ROS/DDS 应用程序，或部署到 FPGA、GPU 和任何 Linux 机器。



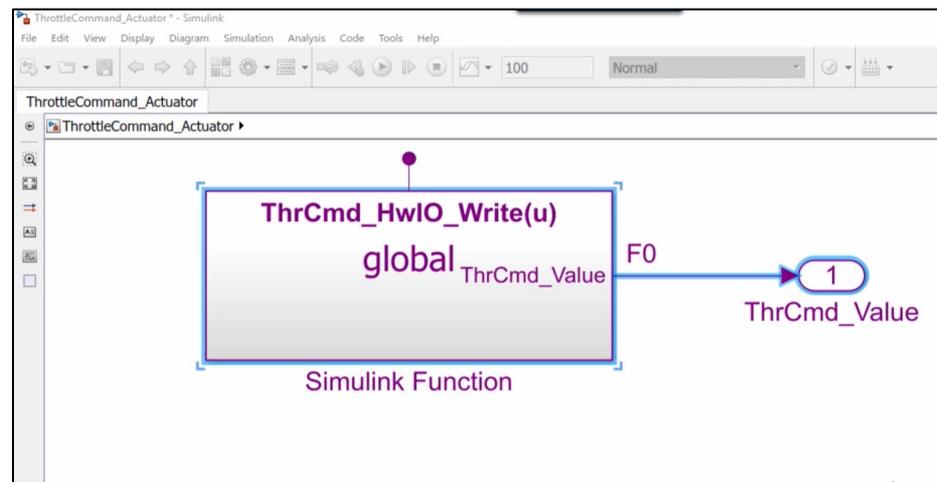
Simulink 可帮助您设计面向服务的应用程序，并部署与行业标准（包括 AUTOSAR Adaptive、ROS 和 DDS）兼容的 C++ 代码。

Simulink 支持面向服务的架构

使用 Simulink 进行基于 SOA 的服务建模

Simulink 提供了一系列建模抽象来帮助增强您的算法模型，使其适合映射到基于标准框架提供的 SOA 的运行时调度和通信。例如，您可以：

- 使用特定的建模风格为运行时调度进行设计分区
- 使用总线捕获软件接口
- 使用 Simulink 函数进行软件服务建模
- 使用调度编辑器和 Stateflow® 来创建简单的测试用例并仿真组合体
- 使用消息绑定到通信中间件

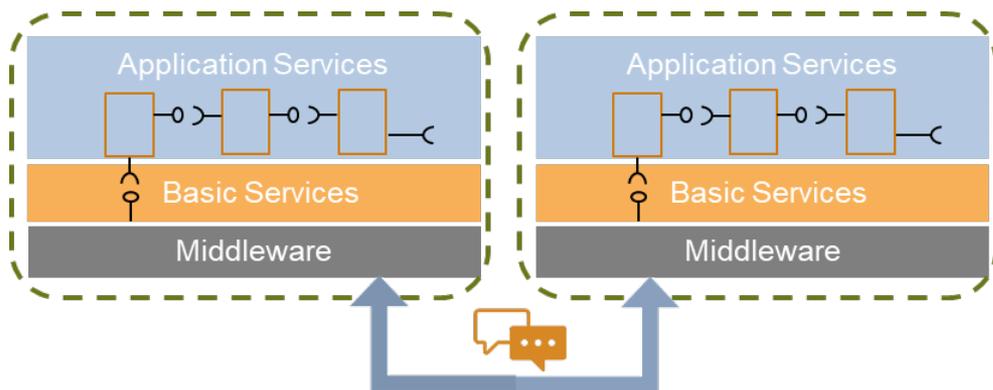


[>> 如何使用 Simulink 函数进行软件服务建模 \(3:07\)](#)

Simulink 支持面向服务的架构

使用 Simulink 仿真基于 SOA 的服务

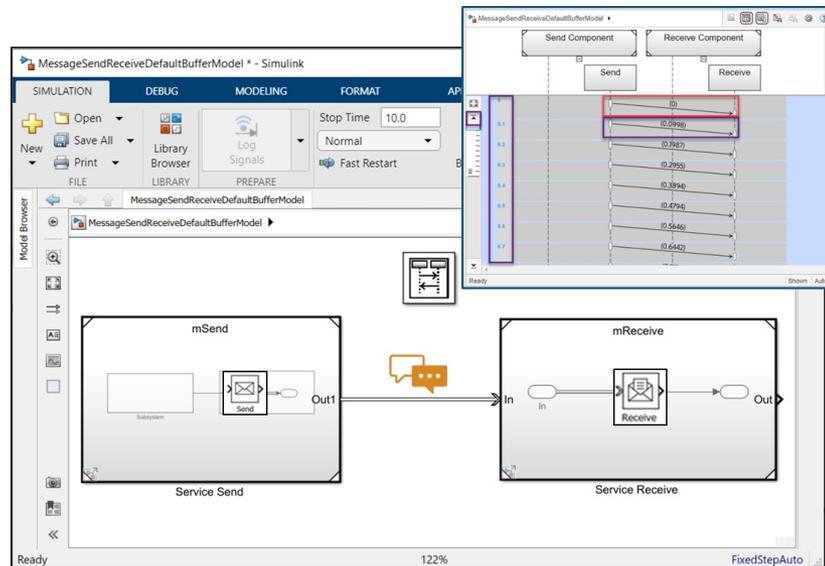
基于 SOA 的应用服务是通过中间件发送和接收消息进行通信的。



了解如何使用库模块构建生成和接收消息的简单模型，并使用事件记录、消息动画和时序查看器对其进行调试。

[>> Simulink 中的消息概述 \(2:25\)](#)

使用 Simulink 消息，您可以通过基于消息的通信对软件组件进行建模。您可以通过模型的根输入和输出端口发送和接收消息。



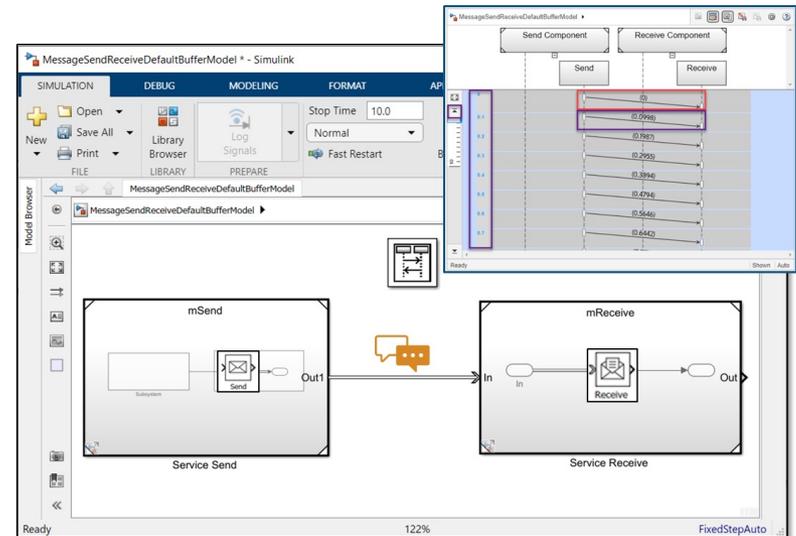
Simulink 支持面向服务的架构

使用 C++ 部署 SOA

从 Simulink 中建模的软件服务生成基于 SOA 的 C 和 C++ 应用程序代码，以便进行部署。

```
// Forward declaration
class MessageSendReceiveDefaultBufferModelModelClass;
class
  MessageSendReceiveDefaultBufferModelModelClassMessa_ReceiveComponent_RecvDataT
  : public RecvData_doubleT
{
private:
  MessageSendReceiveDefaultBufferModelModelClass & aProvider;
public:
  MessageSendReceiveDefaultBufferModelModelClassMessa_ReceiveComponent_RecvDataT
    (MessageSendReceiveDefaultBufferModelModelClass & aProvider);
  virtual void RecvData(real_T *data, int32_T *status);
};

class
  MessageSendReceiveDefaultBufferModelModelClassMessa_ReceiveComponent_SendDataT
  : public SendData_doubleT
{
private:
  MessageSendReceiveDefaultBufferModelModelClass & aProvider;
public:
  MessageSendReceiveDefaultBufferModelModelClassMessa_ReceiveComponent_SendDataT
    (MessageSendReceiveDefaultBufferModelModelClass & aProvider);
  virtual void SendData(const real_T *data, int32_T *status);
};
```



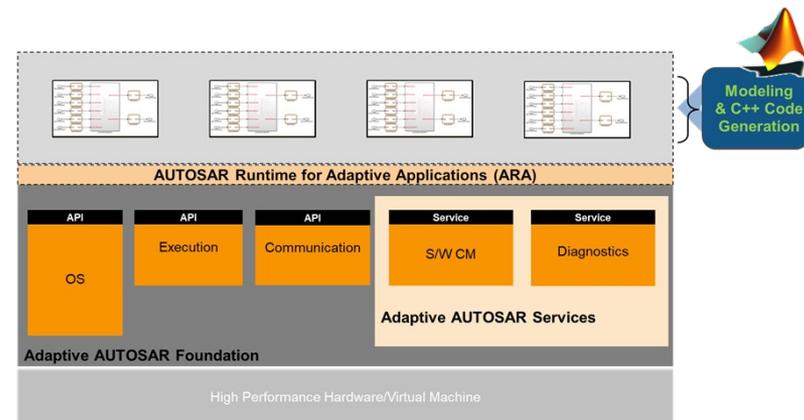
```
// Constructor
MessageSendReceiveDefaultBufferModelModelClass::
  MessageSendReceiveDefaultBufferModelModelClass():
  ReceiveComponentRecvData(*this)
  , SendComponentSendData(*this)
  , Receive_ComponentMDL0B70(get_ReceiveComponentRecvData())
  , Send_ComponentMDL0B71(get_SendComponentSendData())
  , MessageSendReceiveDefaultBuff_M()
{
  // Currently there is no constructor body generated.
}
```

>> 生成 C++ 消息以在 Simulink 组件之间进行通信

Simulink 支持AUTOSAR Adaptive

AUTOSAR Adaptive 平台为汽车行业实现了Adaptive应用程序的 AUTOSAR 运行时(ARA)。

AUTOSAR Blockset 提供 App、模块和字典，用于在 Simulink 中为 AUTOSAR Adaptive 应用程序进行建模、仿真和生成 C++ 代码。



>> [Adaptive 软件组件建模](#)

>> [Simulink 支持 AUTOSAR Adaptive 应用开发 \(23:16\)](#)

AUTOSAR Blockset
设计与仿真 AUTOSAR 软件

The screenshot shows a Simulink model titled 'AUTOSAR Adaptive in Action'. The model includes a 'LaneChangeAlgorithm' block and several event-based components like 'leftLaneDistance', 'leftTurnIndicator', 'rightLaneDistance', and 'rightTurnIndicator'. A presenter is visible on the right side of the frame, holding a device. The video player interface at the bottom shows a timestamp of 19:50 / 23:16 and playback controls.

Simulink 支持AUTOSAR Adaptive

Elektrobit 案例

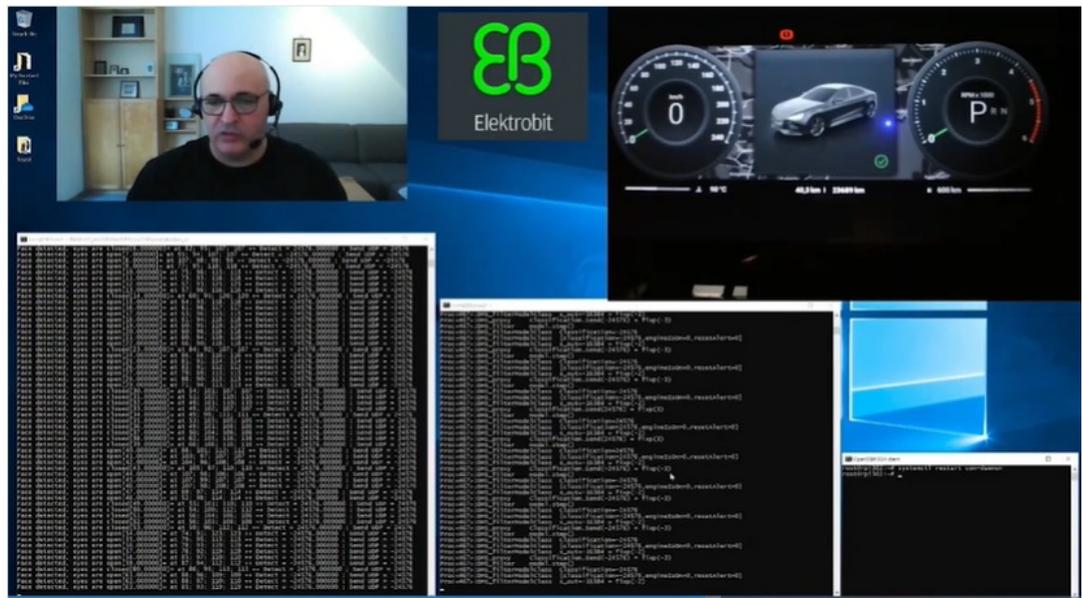
Elektrobit 使用基于模型的设计开发了概念验证的 AUTOSAR Adaptive 驾驶员监控系统。

此用户案例强调了使用 MathWorks 工具开发这种复杂应用程序的简易性。

“使用基于模型的设计开发驾驶员监控系统”

[>> 文章](#)

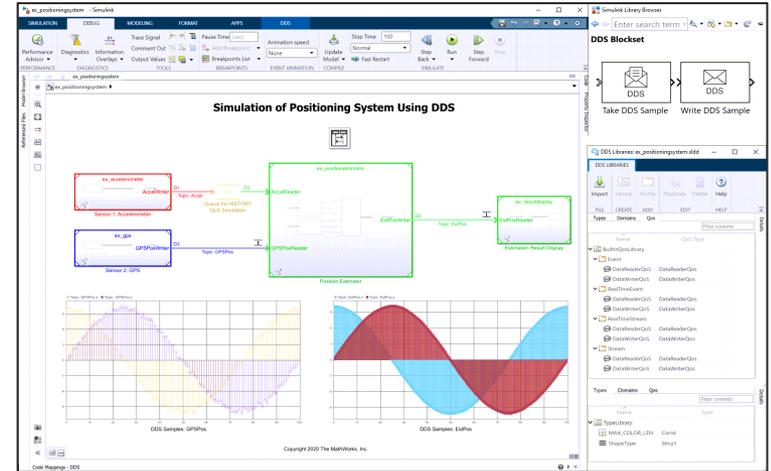
[>> 演示 \(20:50\)](#)



Simulink 支持数据分发服务 (DDS)

DDS 使用发布/订阅通信模式来创建一个去中心化、架构独立、可扩展的异步网络。它非常适合实时应用中的生产质量数据通信。

DDS Blockset 提供 App、字典和模块，用于向 DDS 发布和订阅样本，包括相应的服务质量 (QoS)。它完全集成了 RTI Connex DDS 和 eProsima Fast DDS 堆栈。DDS Blockset 可以从 Simulink 模型生成 C++ 代码和 XML 文件。



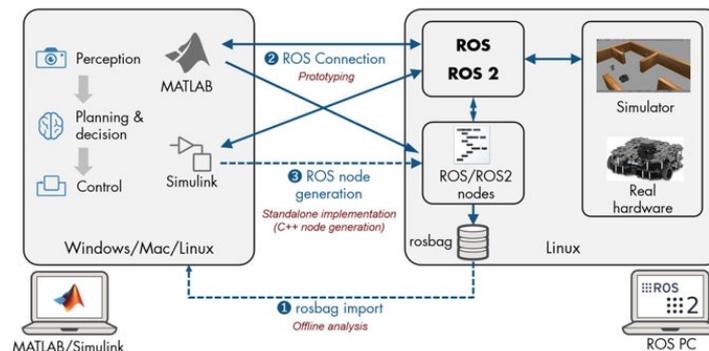
[>> DDS Blockset](#)
[>> 将 DDS 中间件与 Simulink 结合使用 \(11:32\)](#)

DDS Blockset
设计和仿真 DDS 应用

Simulink 支持机器人操作系统 (ROS)

ROS 是一种通信接口，它使一个机器人系统的不同节点能够发现彼此并通过 ROS 网络发送和接收数据。

ROS Toolbox 提供了一个函数库，使您能够与支持 ROS 的物理机器人或机器人仿真器（如 Gazebo®）交换数据。它支持 C++ 代码生成，以从模型中自动生成 ROS 节点并部署到仿真硬件或真实硬件上。



[>> ROS Toolbox](#)

[>> ROS Toolbox 概述 \(2:04\)](#)

ROS Toolbox

设计、仿真和部署基于 ROS 的应用

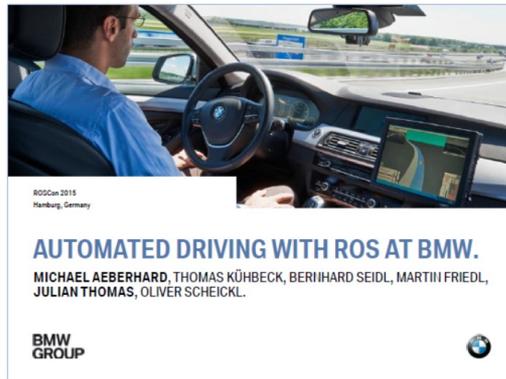
Simulink 支持机器人操作系统 (ROS)

宝马和 Voyage 案例

宝马的 ROS 自动驾驶系统

>> [会议幻灯片](#)

>> [视频 \(28:29\)](#)



USING MATLAB/SIMULINK WITH ROS.

- MathWorks released the Robotics System Toolbox this year for ROS integration with Matlab/Simulink.
- Easily read and analyze data from ROS Bags → useful for evaluating the system.
- Some of our software is implemented as a Simulink model.
 - Use the Toolbox to easily integrated this software into the ROS eco-system:

Michael Aeberhard, BMW Group Research and Technology
http://www.mathworks.com/products/robotics/
Page 13

借助 MATLAB 和 Simulink, Voyage 使用基于模型的设计为自动驾驶出租车开发了纵向控制。他们使用 ROS Toolbox 为感知、运动规划和控制系统构建中间件。

因此, 开发速度提高了三倍, 并且轻松实现了与开源软件的集成。

>> [阅读文章](#)



了解更多

了解有关 **Simulink** 支持 **SOA** 的详细信息

观看

- [使用 Simulink 设计和部署面向服务的架构 \(SOA\)](#)
- [运行时软件建模 \(系列视频\)](#)

阅读

- [SOA 概述](#)

浏览示例

- [软件组件之间基于消息的通信建模](#)
- [使用 Simulink 开发AUTOSAR Adaptive 快速入门](#)
- [使用 Simulink 开发DDS 快速入门](#)
- [使用 Simulink 开发ROS 快速入门](#)