MATLAB EXPO 2019

Systems Engineering
Requirements to Architecture to Simulation

Gaurav Dubey, MathWorks
Key Takeaways

- Digital thread providing traceability between requirements, architecture, and design

---

REQ 3.1 ENABLING CRUISE CONTROL
Cruise control is enabled when

---

ENABLE SWITCH DETECTION
If the Enable switch is pressed

---

Implemented By

---

Implemented By

---

Derives

---

Implemented By

---
Key Takeaways

- Digital thread providing traceability between requirements, architecture, and design
- Connected environment for designing and analyzing architectures and designs
Key Takeaways

- Digital thread providing traceability between requirements, architecture, and design

- Connected environment for designing and analyzing architectures and designs

- Integrated platform for analyzing all parts of your architecture in one multi-domain environment

Dynamic Systems  State Machines  Discrete-Event  Physical Modeling
What does that mean?

**Early in the Process**
Concepts/Descriptions

**Later in the Process**
Models
Early in the Process
Concepts/Descriptions

Later in the Process
Models

What is the Gap?

Digital Thread
Connected Environment
Analysis & Simulation Platform
What goes into the bridge?

Be Intuitive
Facilitate Analysis
Tackle Complexity
Enable Implementation

Digital Thread for Traceability

1. Functional Requirements

1.1. Normal Mode of Operation
During the normal mode of operation, the Fault Tolerant Fuel Control System shall determine the fuel rate which is injected at the valves.

1.1.1. Stoichiometric mixture ratio
During normal model of operation, the System shall maintain the stoichiometric mixture target ratio of 14.6.

1.1.2. Oxygen Sensor (EGO)
MathWorks Solution: System Composer R2019a and

- Be Intuitive
- Facilitate Analysis
- Tackle Complexity
- Enable Implementation

Requirements Coverage Reporting and Impact Analysis

Simulink Requirements
Now let’s see it in action
How to run the DeHaviland model:
1. See the Aerospace Blockset User's Guide for instructions to set up FlightGear.
2. Install the dhc2 geometry model to FlightGear's data/aircraft directory. The geometry is downloadable from www.flightgear.org.
3. To start FlightGear, create an empty model and an initialized batch.

Requirements - UAS_reference_architecture

<table>
<thead>
<tr>
<th>Index</th>
<th>Summary</th>
<th>Implemented</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2</td>
<td>Communications</td>
<td></td>
</tr>
<tr>
<td>1.3</td>
<td>Payload Capabilities</td>
<td></td>
</tr>
<tr>
<td>1.4</td>
<td>Construction</td>
<td></td>
</tr>
<tr>
<td>1.4.1</td>
<td>Modularity</td>
<td></td>
</tr>
<tr>
<td>1.4.2</td>
<td>Propulsion Power</td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>Flying Qualities</td>
<td></td>
</tr>
</tbody>
</table>

100%
Gas Engine: Nine-cylinder, air-cooled, radial aircraft engine
Fuel type: 80/87 grade aviation gasoline
Dry weight: 290 kg
Power output: 400 hp (298 kW) at
Requirement: #35: Propulsion Power

**Summary:** Propulsion Power

**Description:**
- Engine: Nine-cylinder, air-cooled, radial aircraft engine
- Fuel type: 80/87 grade aviation gasoline
- Dry weight/lb (1.03 kW/kg): 290 kg
- Power output: 400 hp (298 kW) at 2,200 RPM up to 6,000 ft (1,800 m)

**Keywords:**
Simulink Requirements
Digital Thread from Requirements to Architecture and Design

Author requirements or view from external source

Link requirements, architectures, design, code and test

Identify gaps in architecture or design

Identify impact of requirement changes
System Composer
Intuitively design system and software architectures

Description

==

Architecture
System Composer

Perform trade studies based on data driven analysis to optimize architectures

- Add custom data
- Create analysis model
- Calculate mass roll-up data
System Composer
Tackle Architecture complexity with spotlight views

Composition

Spotlight

MATLAB EXPO 2019
System Composer
System and software architectures connected to implementations in Simulink

Generate Simulink models from architecture components

Link Simulink models to architecture components
Simulink: A Multi-Language Simulation Environment

Dynamic Systems
State Machines
Discrete-Event Systems

Physical Modeling
Object-Oriented
Learn More

- Simulink Requirement Webpage
- System Composer Webpage
- System Modeling and Simulation Webpage

- Trial