OPEN STPA WITH RAAML AND GAPHOR

2021 STAMP Workshop

Dan Yeaw & Kyle Post
A VIBRANT COMMUNITY

We want to build an open community for systems safety

• A low barrier to join, learn, and grow 🚶
• Diverse contributors who want to help 🛠️
• Built on open standards and open source software
SAFETY STANDARDS

- Each industry has developed domain specific standards, most derived from IEC 61508
- New techniques, like STPA, can improve how we do safety analysis
- However, may lack rigor without a standardized and consistent language and automation
RAAML

- Precise language for systems safety
- OMG spec 1.0 beta available, final release soon
- STPA metamodel library based on the STPA Handbook
- Facilitates exchange of info between tools and organizations
An open source UML, SysML, and now RAAML modeling tool written in Python
Fast and easy to use, while still having a full data model
Improves rigor through consistency, helps add automation
Step 1: Define Purpose of the Analysis
Step 2: Model the Control Structure
Step 3: Identify Unsafe Control Actions

- UnsafeControlAction: NotProvided (from Part of Profile)
- UnsafeControlAction: Provided (from Part of Profile)
- UnsafeControlAction: Early (from Part of Profile)
- UnsafeControlAction: Late (from Part of Profile)
- UnsafeControlAction: OutOfSequence (from Part of Profile)
- UnsafeControlAction: TooShort (from Part of Profile)
- UnsafeControlAction: TooLong (from Part of Profile)

- UnsafeControlAction: BSCU Autobrake does not provide the Brake control action during landing roll when the BSCU is armed
- UnsafeControlAction: BSCU Autobrake provides Brake control action during a normal takeoff
- UnsafeControlAction: BSCU Autobrake provides Brake control action with an insufficient level of braking during landing roll
- UnsafeControlAction: BSCU Autobrake provides Brake control action with directional or asymmetrical braking during landing roll
- UnsafeControlAction: BSCU Autobrake provides the Brake control action too late (> TBD seconds) after touchdown
- UnsafeControlAction: BSCU Autobrake stops providing the Brake control action too early (before TBD taxi speed attained) when aircraft lands

- ControlAction: Brake Cmd (from Tigrak)
Step 4: Identify Loss Scenarios

Context: The BSCU is armed and the aircraft begins landing roll.

**Situation**

**Loss Scenario**

(From root of tree)

**Process Flow**
The BSCU does not provide the Brake control action.

**Cause Factor**
This flawed process model will occur if the received feedback momentarily indicates zero speed during landing roll. The received feedback may momentarily indicate zero speed during anti-skid operation, even though the aircraft is not stopped.

Context: The BSCU is armed and the aircraft begins landing roll.

**Situation**

**Loss Scenario**

(From root of tree)

**Process Flow**
The BSCU incorrectly believes the aircraft has already come to a stop.

**Cause Factor**
This flawed process model will occur if the touchdown indication is not received upon touchdown.
For more information see:

https://omg.org/spec/RAAML

https://gaphor.org