THE SAFETY ANNEX FOR THE ARCHITECTURE ANALYSIS AND DESIGN LANGUAGE

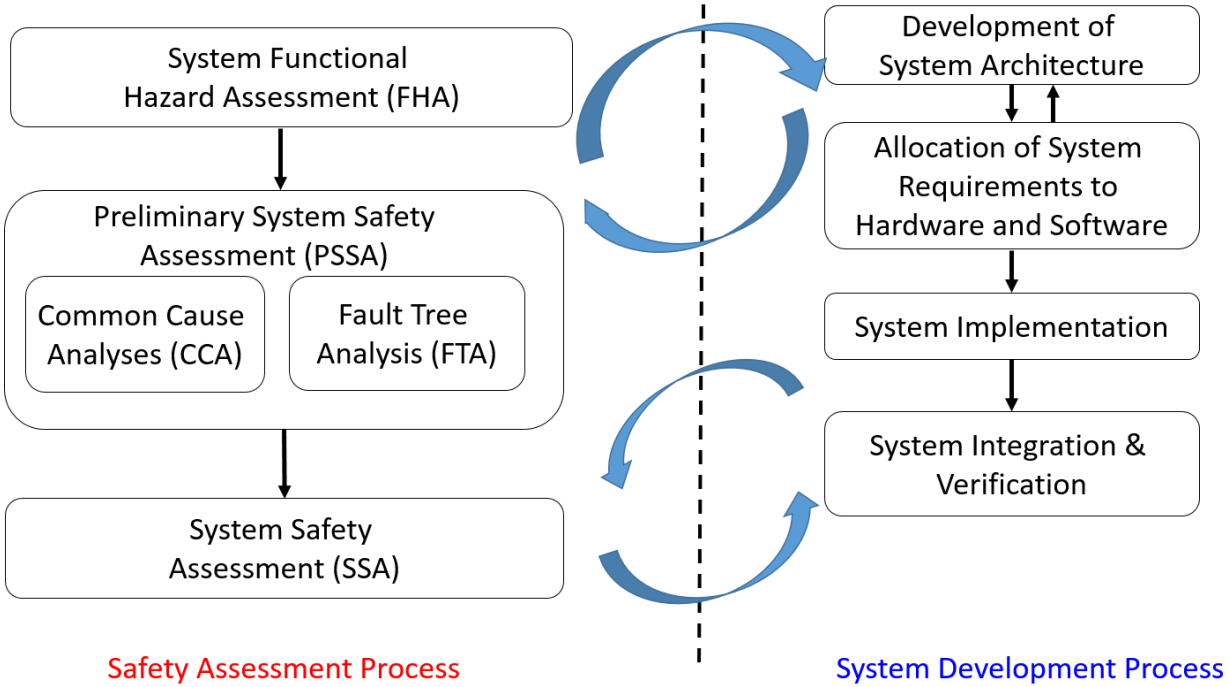
EMBEDDED REAL TIME SYSTEMS
JANUARY 2020

Danielle Stewart, Jing (Janet) Liu, Darren Cofer,
Mats Heimdahl, Michael Whalen, Michael Peterson

Collins Aerospace

University of Minnesota

TRADITIONAL SAFETY ASSESSMENT PROCESS

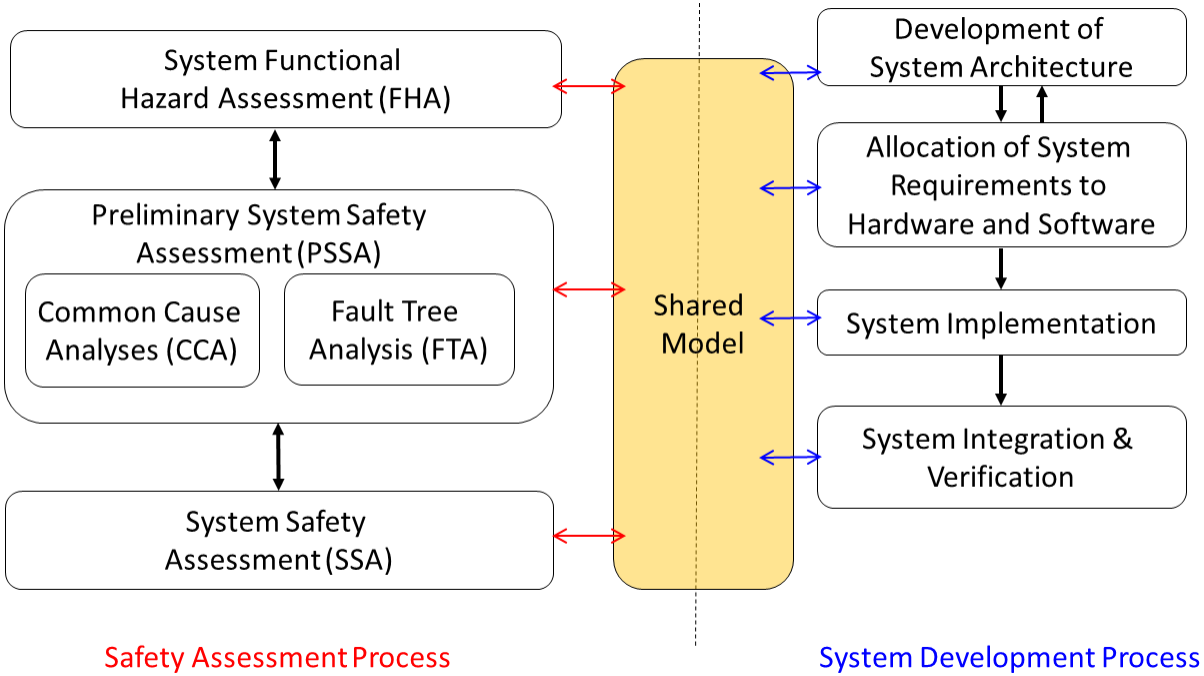


- Completeness of the traditional system safety analysis artifacts is subjective and dependent on the skill of the practitioner
- Based on informal (or non-existent) system models that are incomplete, imprecise, possibly inconsistent
- Architectural details about the system behavior gathered from multiple sources
- Developing adequate understanding especially for software components is a difficult and time consuming endeavor

Collins Aerospace

University of Minnesota

MBSA PROCESS

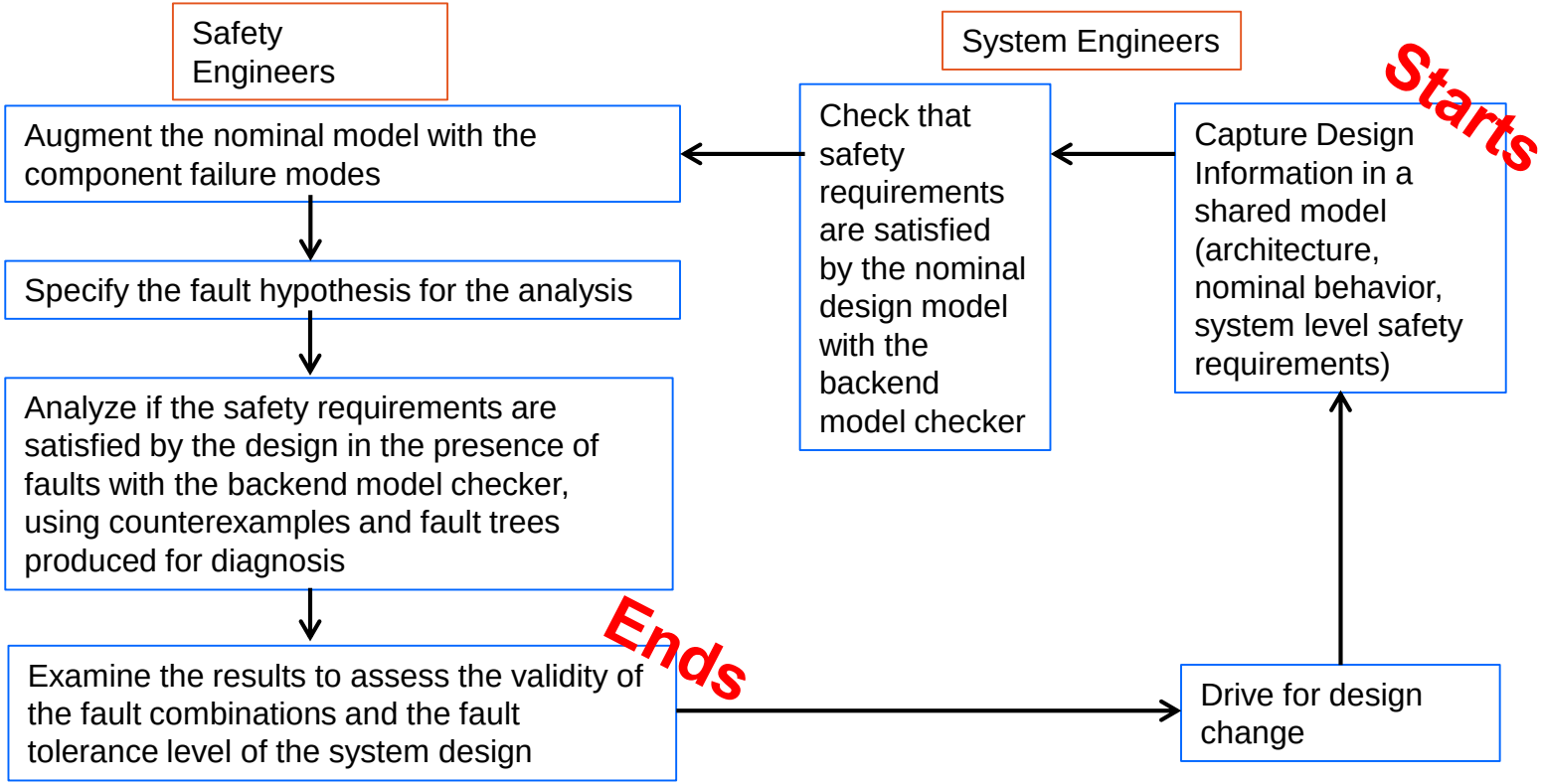


- Have system developers and safety engineers use the same system models created during a model-based development process
- Extend system model to add capabilities for reasoning about faulty behaviors

Collins Aerospace

University of Minnesota

MBSA PROCESS STEPS



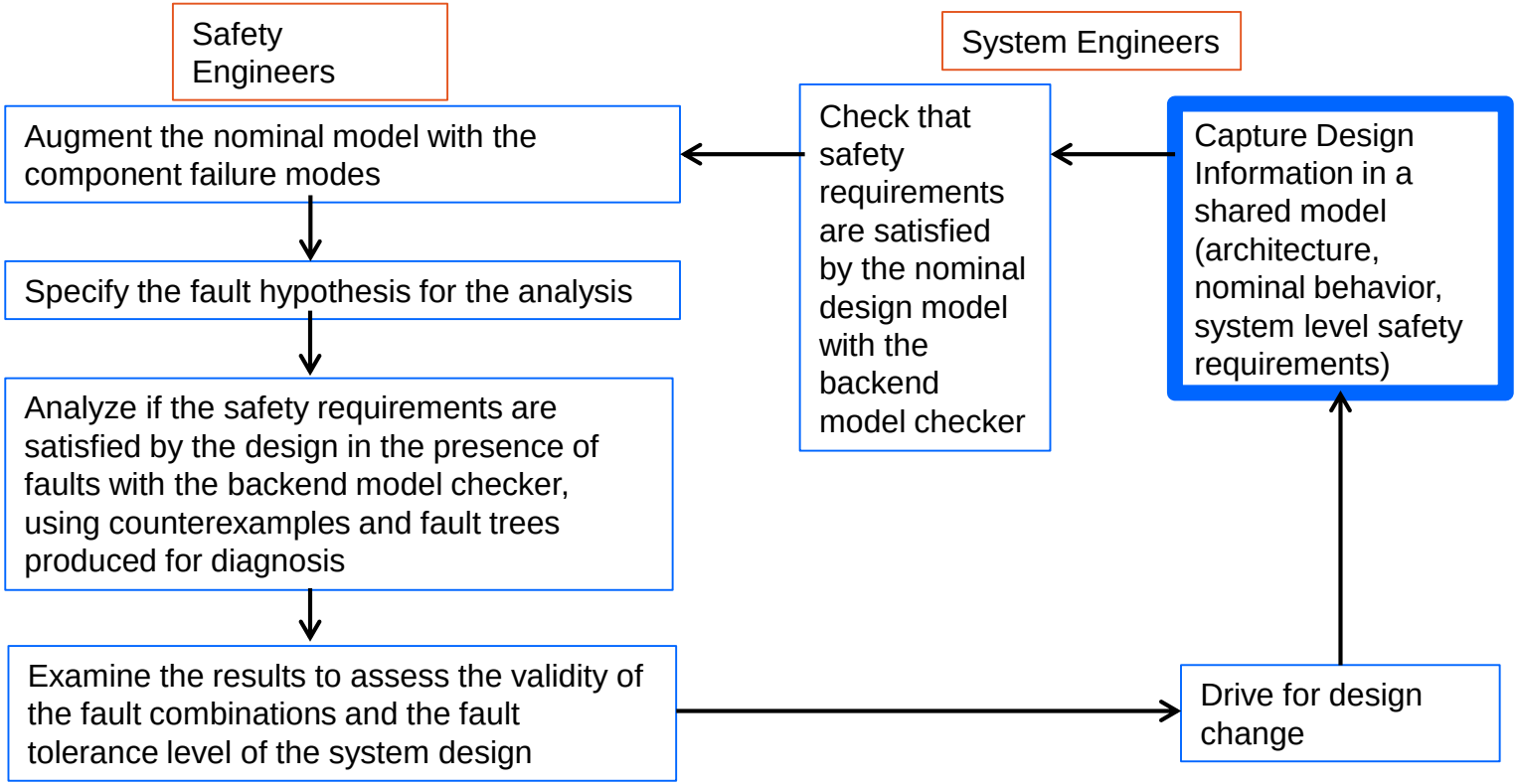
Collins Aerospace

University of Minnesota

WHAT WILL HELP SAFETY ANALYSTS DO THIS?

- Shared model
  - Modeling language expressive enough to capture HW/SW, standard language
- Flexible error propagations
  - Behavioral AND explicit propagations
- Flexible fault modeling
  - Symmetric, asymmetric, dependent, independent
- Backend model checker
  - Used to assess design with or without active faults
- Ability to generate assessment artifacts
  - System traces, counterexamples, minimal cut sets, fault trees, etc.

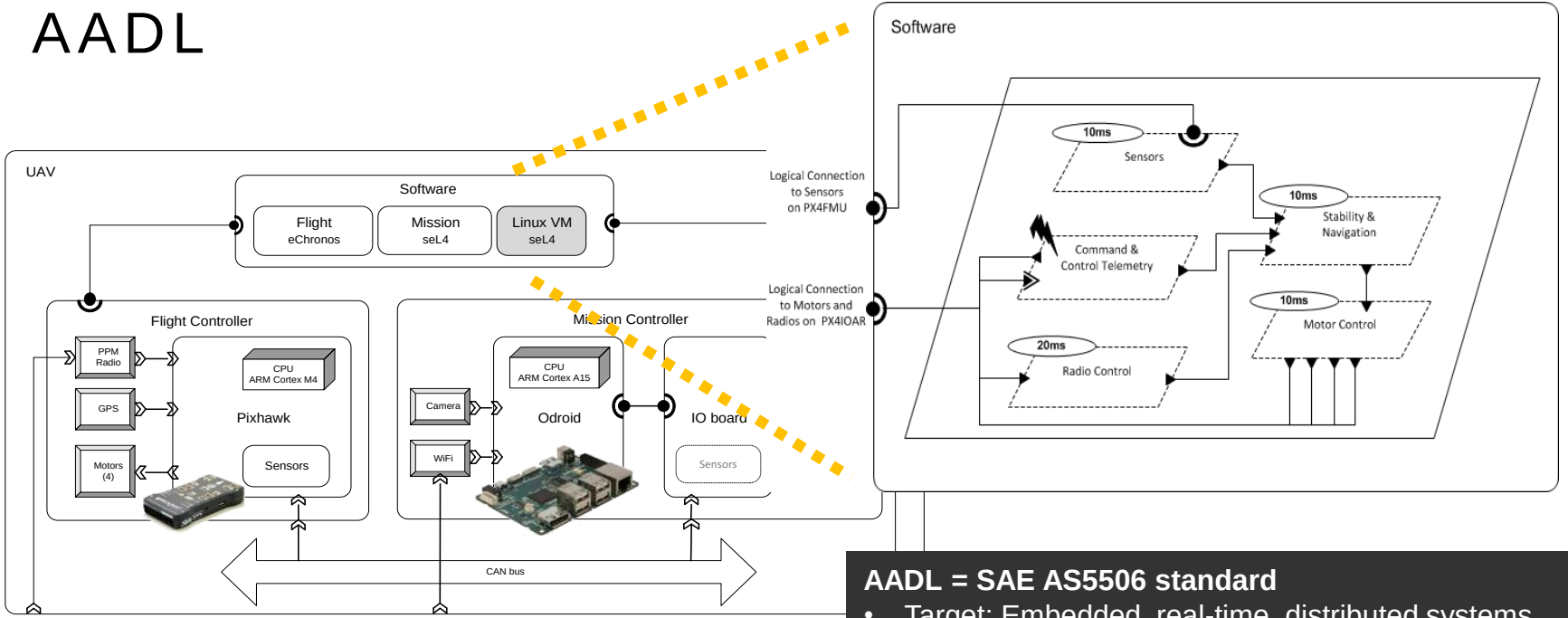
MBSA PROCESS STEPS



Collins Aerospace

University of Minnesota

AADL



AADL = SAE AS5506 standard

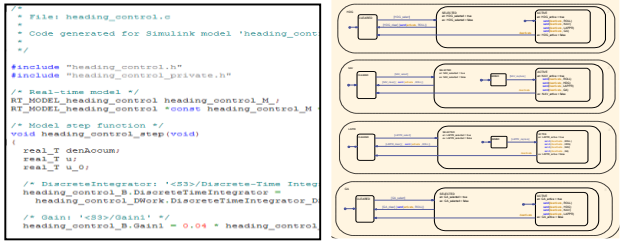
- Target: Embedded, real-time, distributed systems
- Describes both hardware and software
- Extensible syntax (annex)
- Open source tools, supported by SEI

Collins Aerospace

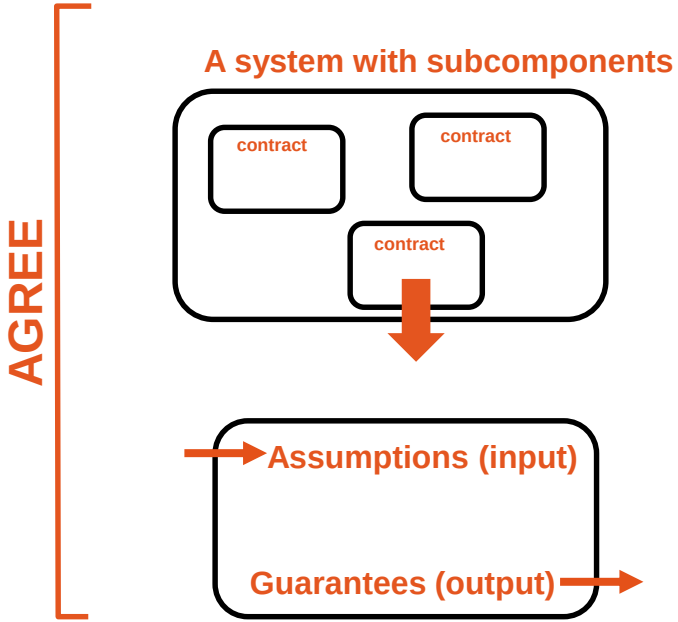
University of Minnesota

ASSUME-GUARANTEE REASONING ENVIRONMENT (AGREE)

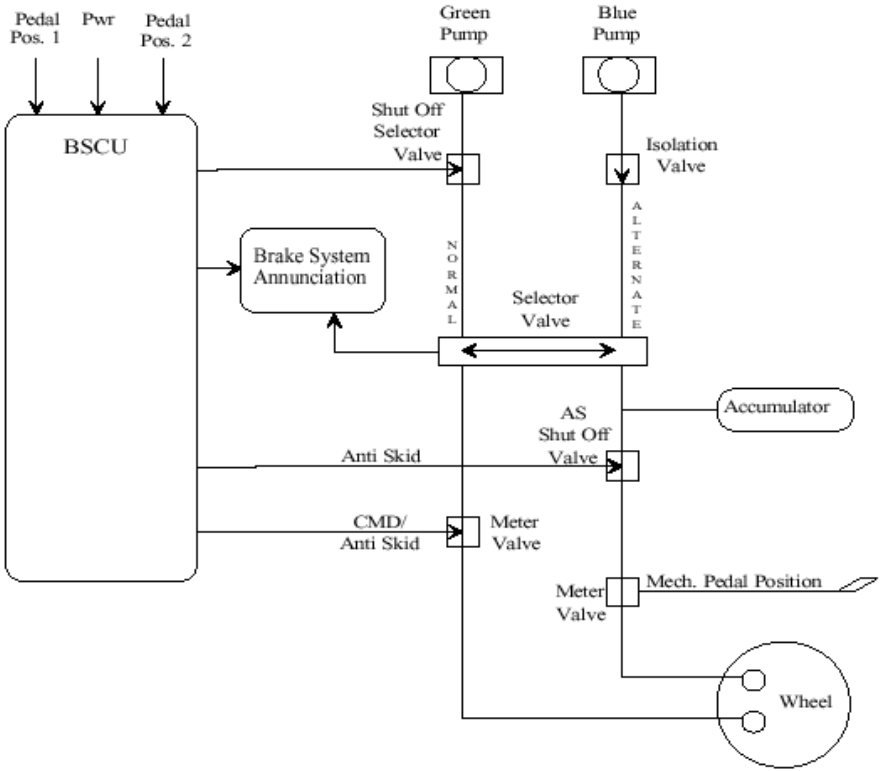
Component Implementation



- Each component has a contract consisting of assumptions and guarantees
  - Assumptions: Constraints over what a component expects from its environment
  - Guarantees: Constraints over how a component behaves in response to its environment (requirements)



WHEEL BRAKE SYSTEM (WBS)



Collins Aerospace

University of Minnesota

AADL AND AGREE

AGREE is incorporated as an **annex** of AADL

```
system SensorPedalPosition
    features
        -- Input
        mech_pedal_pos : in data port Base_Types::Boolean;
        -- Output
        elec_pedal_pos : out data port Base_Types::Boolean;

    annex agree {**
            guarantee "(SensorPedalPosition) mechanical pedal position is
                        equivalent to electrical pedal position." :
                true -> (mech_pedal_pos <=> elec_pedal_pos);
    **};
end SensorPedalPosition;
```

SAFETY REQUIREMENTS FOR WBS

S18-WBS-R-0321 Loss of all wheel braking during landing or RTO shall be less than 5.0×10^{-7} per flight.
S18-WBS-R/L-0322 Asymmetrical loss of wheel braking (Left/Right) shall be less than 5.0×10^{-7} per flight.
S18-WBS-0323 Never inadvertent braking with all wheels locked shall be less than 1.0×10^{-9} per takeoff.
S18-WBS-0324 Never inadvertent braking with all wheels shall be less than 1.0×10^{-9} per takeoff.
S18-WBS-0325-wheelX Never inadvertent braking of wheel X shall be less than 1.0×10^{-9} per takeoff. .

Collins Aerospace

University of Minnesota

MBSA PROCESS STEPS

Safety Engineers

Augment the nominal model with the component failure modes

Specify the fault hypothesis for the analysis

Analyze if the safety requirements are satisfied by the design in the presence of faults with the backend model checker, using counterexamples and fault trees produced for diagnosis

Examine the results to assess the validity of the fault combinations and the fault tolerance level of the system design

System Engineers

Check that safety requirements are satisfied by the nominal design model with the backend model checker

Capture Design Information in a shared AADL/AGREE model (architecture, nominal behavior, system level safety requirements)

Drive for design change

Collins Aerospace

University of Minnesota

NOMINAL MODEL ANALYSIS OUTPUT

Property	Result
Contract Guarantees	16 Valid
phys_sys assume: (PhysicalSystem) Hydraulic pressure and ground speed bounded between 0 and 10 inclusi	Valid (3s)
ctrl_sys assume: (ControlSystem) Ground speed always greater than zero.	Valid (3s)
Subcomponent Assumptions	Valid (5s)
lemma: (S18-WBS-R-0321) Never loss of all wheel braking	Valid (5s)
lemma: (S18-WBS-R-0322-left) Asymmetrical left braking.	Valid (6s)
lemma: (S18-WBS-R-0322-right) Asymmetrical right braking	Valid (6s)
lemma: (S18-WBS-0323) Never inadvertent braking with all wheels locked.	Valid (6s)
lemma: (S18-WBS-0324) Never inadvertent braking of all wheels.	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 1	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 2	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 3	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 4	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 5	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 6	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 7	Valid (6s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 8	Valid (6s)

MBSA PROCESS STEPS

Safety Engineers

Augment the nominal model with the component failure modes

Specify the fault hypothesis for the analysis

Analyze if the safety requirements are satisfied by the design in the presence of faults with the backend model checker, using counterexamples and fault trees produced for diagnosis

Examine the results to assess the validity of the fault combinations and the fault tolerance level of the system design

System Engineers

Check that safety requirements are satisfied by the nominal design model with the backend model checker

Capture Design Information in a shared AADL/AGREE model (architecture, nominal behavior, system level safety requirements)

Drive for design change

Collins Aerospace

University of Minnesota

NOMINAL MODEL ANALYSIS OUTPUT

Safety syntax is incorporated as an **annex** of AADL

```
system monitor
    features
        craft_input : in data port Base_Types::Boolean;

        craft_response : out data port Base_Types::Boolean;

    annex agree {**
        guarantee "When gps signal fails, monitor responds." :
            (not craft_input) => (craft_response);
    **};

    annex safety {**

        fault Monitor_Failure "Monitor response is inverted" : faults.inverted_fail {
        inputs: val_in <- craft_response;
        outputs: craft_response <- val_out ;
        probability: 1.0E-3 ;
        duration: permanent;
        }

    **};

end monitor;
```

DEFINING FAULTS ON COMPONENT OUTPUTS

- Valves
  - Stuck open
  - Stuck closed
  - Stuck non-deterministically
- Sensors
  - Output inverted
- Pumps
  - Output zero
- Calculating components and Gates
  - Erroneous data

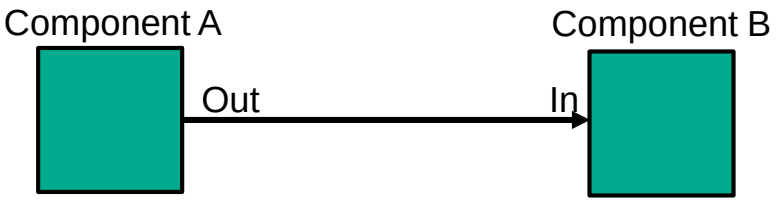
A FAULT DEFINITION

```
fault SelectorValve_Failed "Stuck at last position (blue line)." : faults.fail_to {
    inputs: val_in <- blue_select_out, alt_val <- pre(blue_select_out);
    outputs: blue_select_out <- val_out;
    probability: 1.0E-5 ;
    duration: permanent;
}
```

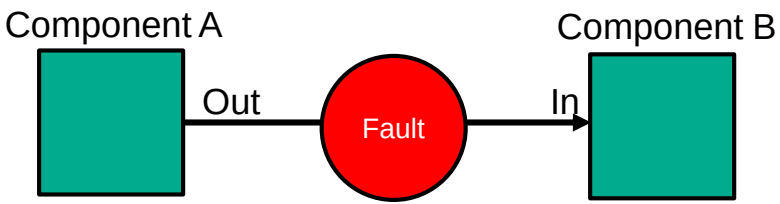
BEHAVIORAL ERROR PROPAGATION

- Wrap nominal component output in fault
- Watch behavior of system through AGREE contracts when fault is activated

Nominal Behavior:



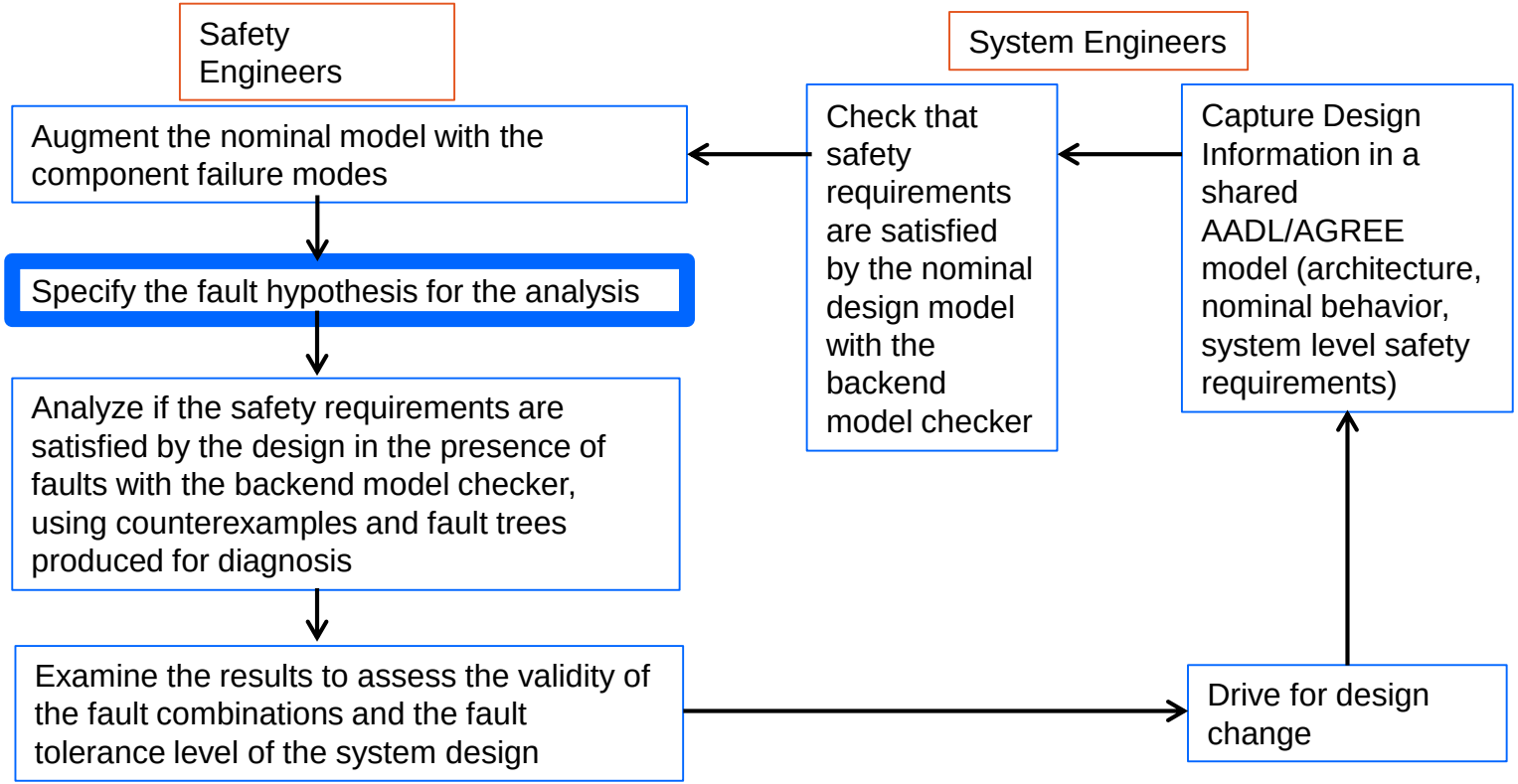
Fault Behavior:



Collins Aerospace

University of Minnesota

MBSA PROCESS STEPS



Collins Aerospace

University of Minnesota

SPECIFY FAULT HYPOTHESIS STATEMENT

Specifies type of analysis to perform

- Max N Analysis

```
annex safety{**
   analyze : max 1 fault
**};
```

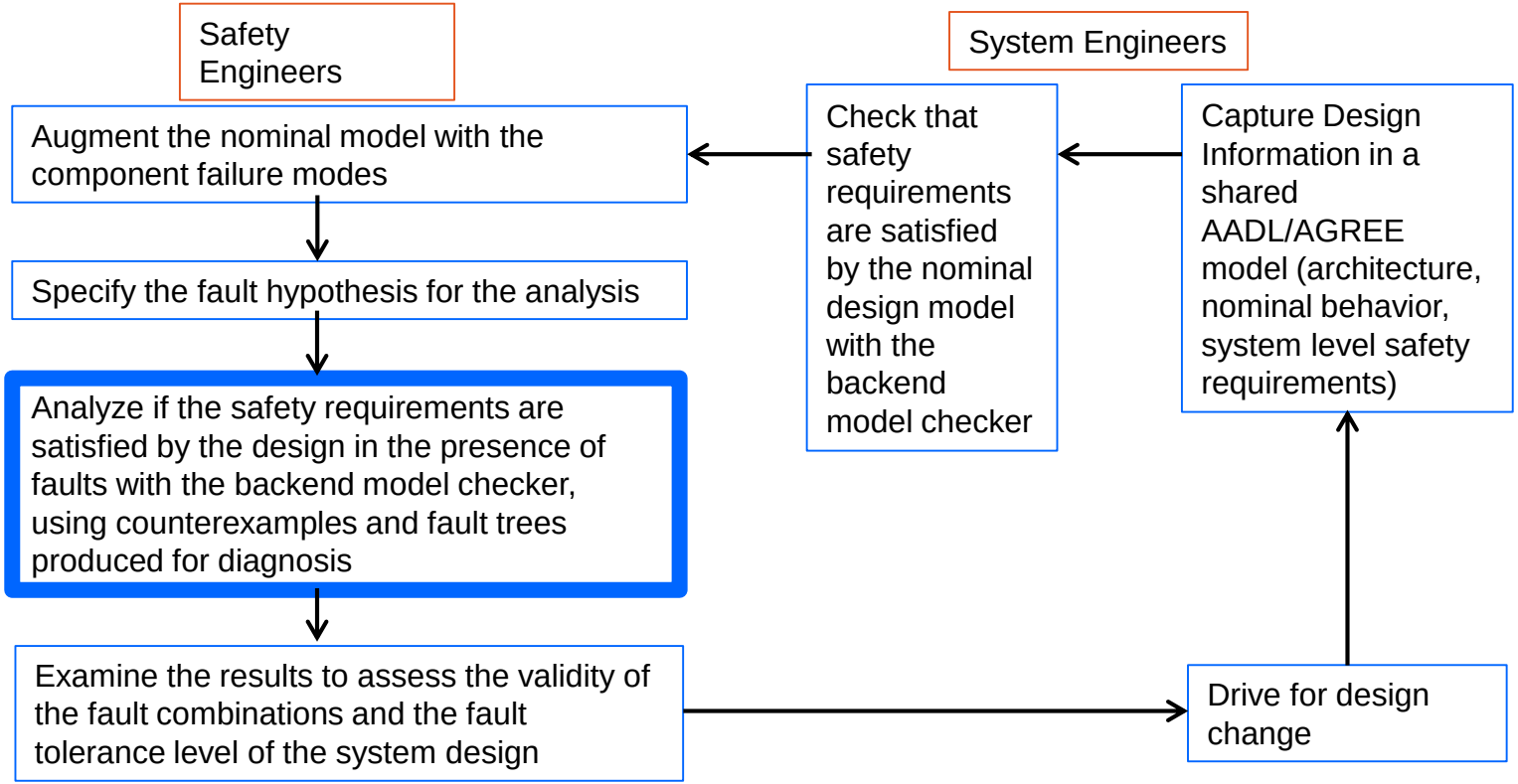
- Probabilistic Analysis

```
annex safety{**
   analyze : probability 1.0E-9
**};
```

Collins Aerospace

University of Minnesota

MBSA PROCESS STEPS



Collins Aerospace

University of Minnesota

FAULT ANALYSIS RESULTS

- Max 1 Fault Analysis for WBS

Property	Result
Verification for WBS.inst	9 Invalid, 21 Valid
Contract Guarantees	9 Invalid, 7 Valid
phys_sys assume: (PhysicalSystem) Hydraulic pressure and ground speed bounded between 0 and	Valid (2s)
ctrl_sys assume: (ControlSystem) Ground speed always greater than zero.	Valid (2s)
Subcomponent Assumptions	Valid (5s)
lemma: (S18-WBS-R-0321) Never loss of all wheel braking	Valid (5s)
lemma: (S18-WBS-R-0322-left) Asymmetrical left braking.	Valid (5s)
lemma: (S18-WBS-R-0322-right) Asymmetrical right braking	Valid (5s)
lemma: (S18-WBS-0323) Never inadvertent braking with all wheels locked.	Invalid (4s)
lemma: (S18-WBS-0324) Never inadvertent braking of all wheels.	Valid (9s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 1	Invalid (4s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 2	Invalid (5s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 3	Invalid (5s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 4	Invalid (5s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 5	Invalid (5s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 6	Invalid (5s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 7	Invalid (5s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 8	Invalid (5s)

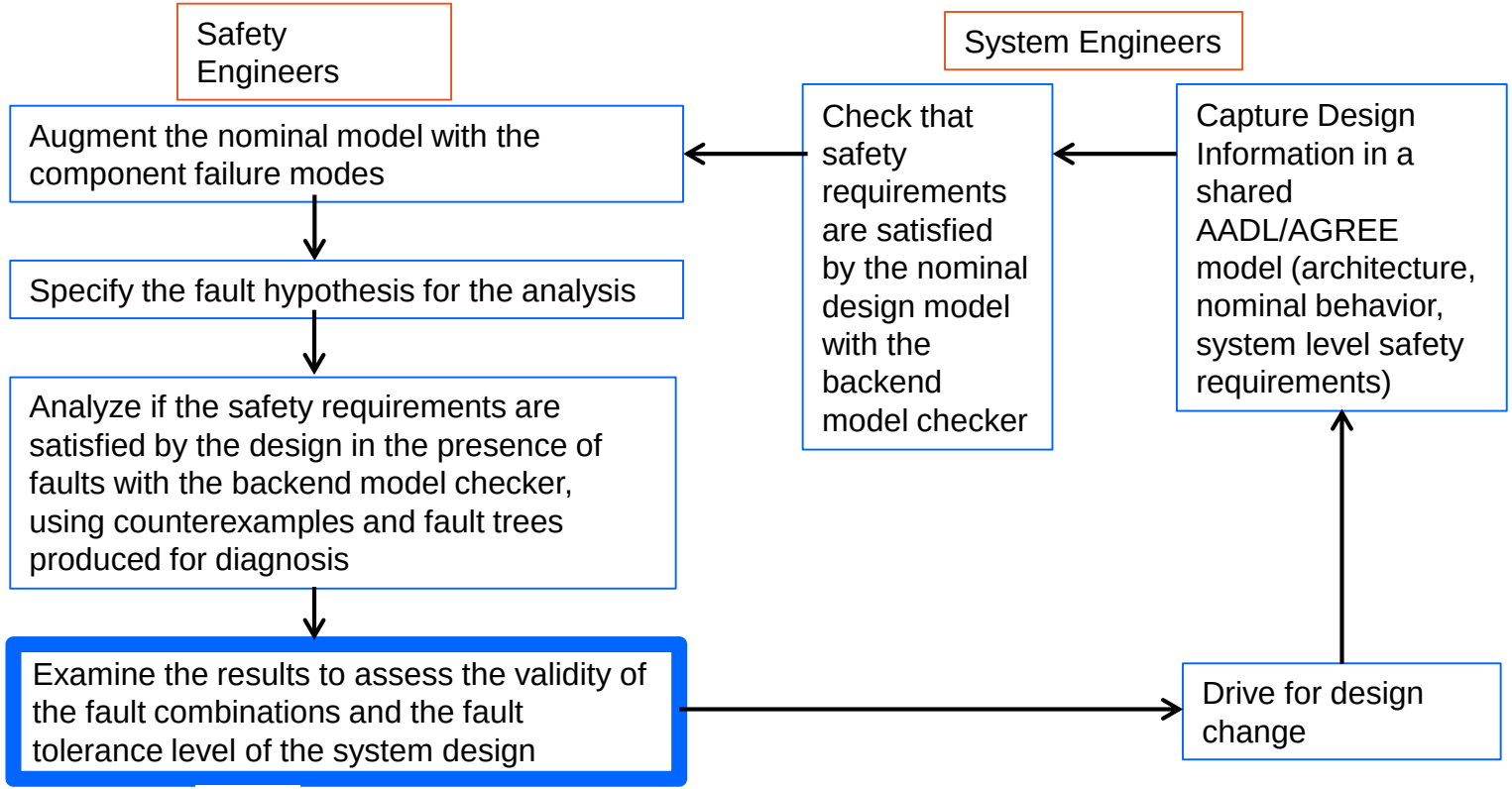
Collins Aerospace

University of Minnesota

OTHER GENERATED ARTIFACTS

- Counterexamples showing trace of the system
- Minimal cut sets
  - Tally format
  - Textual format
- Fault Trees (with or without probabilistic information)
  - Text format
  - Graph format

MBSA PROCESS STEPS



Collins Aerospace

University of Minnesota

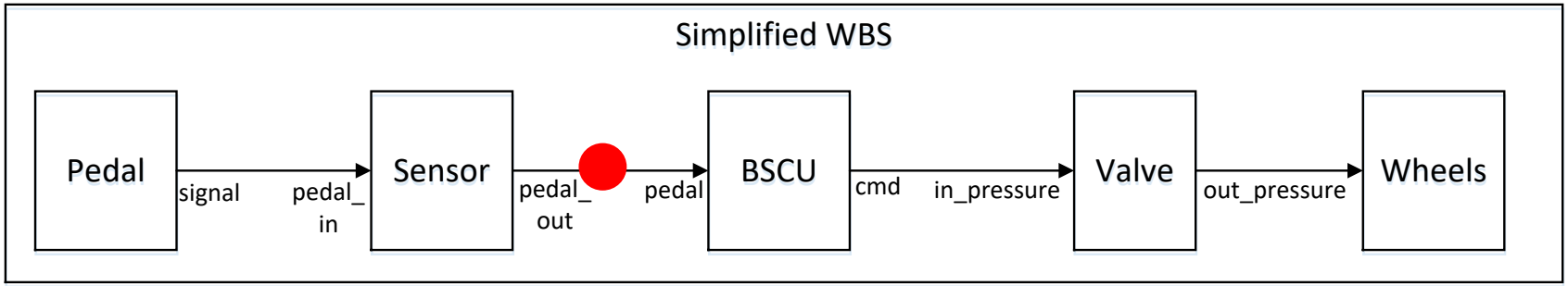
COUNTEREXAMPLE VIEW

- Max 1 Fault Analysis for WBS

lemma: (S18-WBS-0323) Never inadvertent braking with all wheels locked	true	false
(Sensor) Inverted boolean (Erroneous data) fault		
(wheel_sensor1__fault_1)	false	false
(wheel_sensor2__fault_1)	false	false
(wheel_sensor3__fault_1)	false	false
(wheel_sensor4__fault_1)	false	false
(wheel_sensor5__fault_1)	false	false
(wheel_sensor6__fault_1)	false	false
(wheel_sensor7__fault_1)	false	false
(wheel_sensor8__fault_1)	false	false
(SensorPedalPosition) Inverted boolean fault		
(pedal_sensor_L__fault_1)	false	true
(pedal_sensor_R__fault_1)	false	false

Collins Aerospace

ASSESS BEHAVIOR WITH FAULTS



- Pedal not pressed
- Sensor reports that it was pressed
- BSCU commands braking

Collins Aerospace

University of Minnesota

MBSA PROCESS STEPS

Safety Engineers

Augment the nominal model with the component failure modes

Specify the fault hypothesis for the analysis

Analyze if the safety requirements are satisfied by the design in the presence of faults with the backend model checker, using counterexamples and fault trees produced for diagnosis

Examine the results to assess the validity of the fault combinations and the fault tolerance level of the system design

System Engineers

Check that safety requirements are satisfied by the nominal design model with the backend model checker

Capture Design Information in a shared AADL/AGREE model (architecture, nominal behavior, system level safety requirements)

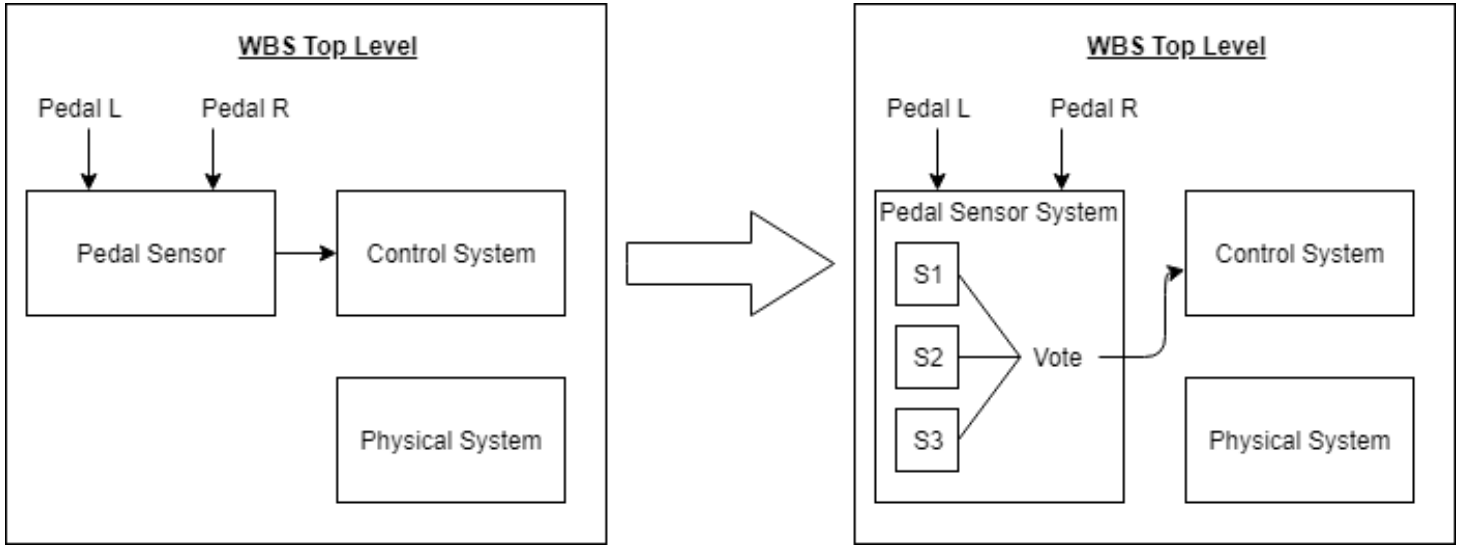
Drive for design change

Collins Aerospace

University of Minnesota

ASSESS BEHAVIOR WITH FAULTS

Redundancy in the Pedal Sensor



RESULTS OF DESIGN CHANGE

Now resilient to this single fault

Property	Result
Contract Guarantees	32 Valid
phys_sys assume: (PhysicalSystem) Hydraulic pressure and ground speed bounded between 0 and	Valid (1s)
ctrl_sys assume: (ControlSystem) Ground speed always greater than zero.	Valid (1s)
Subcomponent Assumptions	Valid (1s)
lemma: (S18-WBS-R-0321) Never loss of all wheel braking	Valid (1s)
lemma: (S18-WBS-R-0322-left) Asymmetrical left braking.	Valid (1s)
lemma: (S18-WBS-R-0322-right) Asymmetrical right braking	Valid (1s)
lemma: (S18-WBS-0323) Never inadvertent braking with all wheels locked.	Valid (1s)
lemma: (S18-WBS-0324) Never inadvertent braking of all wheels.	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 1	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 2	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 3	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 4	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 5	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 6	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 7	Valid (1s)
lemma: (S18-WBS-0325) Never inadvertent braking of wheel 8	Valid (1s)

Collins Aerospace

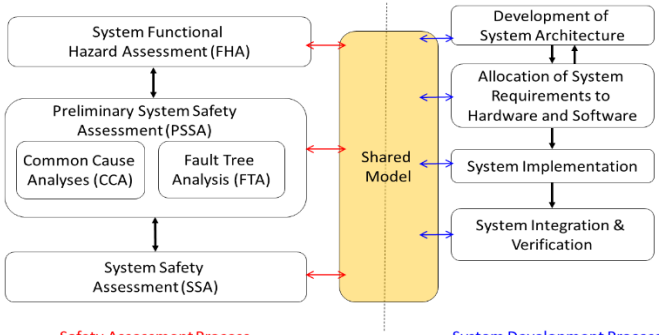
University of Minnesota

FUTURE WORK

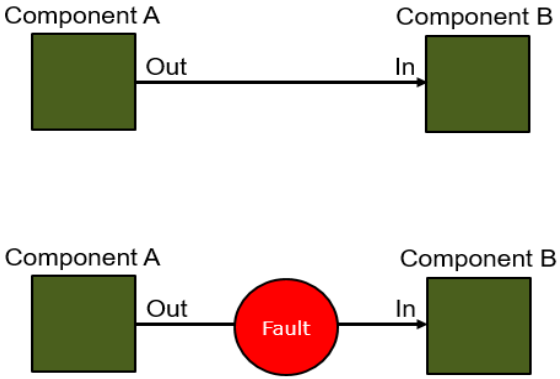
- Use the information gleaned from the collection of minimal cut sets for interesting things:
  - Use fault probabilities to calculate the system threshold
  - Integrate into hierarchy of system to show more meaningful fault trees

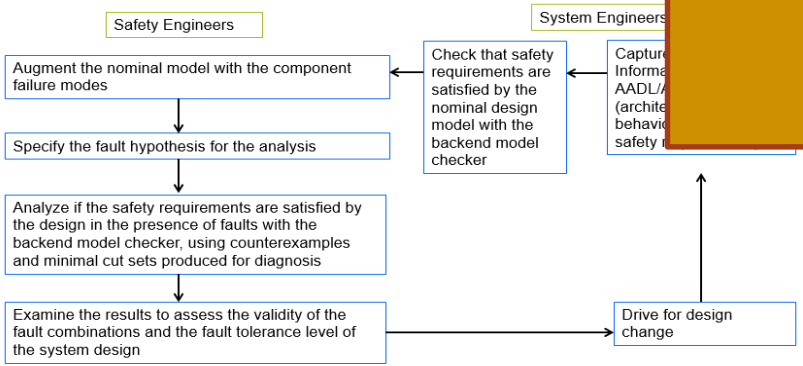
- Plans for use in a large scale aircraft system in 2020

QUESTIONS



Nominal Behavior:





er inadvertent braking with all wheels locked	true	false
oneous data) fault		
	false	false
(wheel_sensor2_fault_1)	false	false
(wheel_sensor3_fault_1)	false	false
(wheel_sensor4_fault_1)	false	false
(wheel_sensor5_fault_1)	false	false
(wheel_sensor6_fault_1)	false	false
(wheel_sensor7_fault_1)	false	false
(wheel_sensor8_fault_1)	false	false
(SensorPedalPosition) Inverted boolean fault		
(pedal_sensor_L_fault_1)	false	true
(pedal_sensor_R_fault_1)	false	false

Collins Aerospace

University of Minnesota

EXAMPLE SAFETY ARTIFACTS GENERATED

Minimal Cut Sets

List

```
Minimal Cut Sets for property violation:
property lustre name: c0123_impl___GUARANTEE1
property description: C0123 out0 value
Total 3 Minimal Cut Sets
Minimal Cut Set # 1
Cardinality 1
original fault name, description: C1_out_negation, "C1 out1 negation fault"
lustre component, fault name: C123, c123_fault__independently__active__C1__C1__fault_1
failure rate, default exposure time: 1.0E-6, 1.0

Minimal Cut Set # 2
Cardinality 1
original fault name, description: C2_out_fail_to_zero, "C2 out2 fail to zero fault"
lustre component, fault name: C123, c123_fault__independently__active__C2__C2__fault_1
failure rate, default exposure time: 1.0E-5, 1.0

Minimal Cut Set # 3
Cardinality 1
original fault name, description: C0_out_fail_to_one, "C0 out0 fail to one fault"
lustre component, fault name: C0123_impl, c0123_impl_fault__independently__active__C0__C0__fault_1
failure rate, default exposure time: 1.0E-7, 1.0
```

Tally

```
Minimal Cut Sets for property violation:
property lustre name: c0123_impl___GUARANTEE1
property description: C0123 out0 value
Total 3 Minimal Cut Sets
Cardinality 1 number: 3
```

Fault Trees

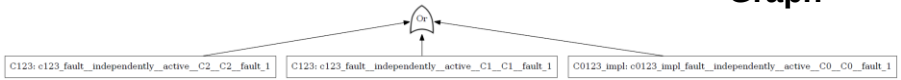
Text

```
#       val c0123_impl___GUARANTEE1 : (string * string) FaultTree.ftree =
  SUM
   [PRO
     [Leaf
       (("C0123_impl",
         "c0123_impl_fault__independently__active__C0__C0__fault_1"),
       1e-07, 1.)];
    PRO
     [SUM
       [PRO
         [Leaf
           (("C123", "c123_fault__independently__active__C2__C2__fault_1"),
           1e-05, 1.)];
        PRO
         [Leaf
           (("C123", "c123_fault__independently__active__C1__C1__fault_1"),
           1e-06, 1.)]]]]
```

Probabilities

```
#     - : (string * string) FaultTree.pexp =
Sum
 [Var
   ("C0123_impl", "c0123_impl_fault__independently__active__C0__C0__fault_1");
  Var ("C123", "c123_fault__independently__active__C1__C1__fault_1");
  Var ("C123", "c123_fault__independently__active__C2__C2__fault_1")]
# - : float * float = (1.10999383951691684e-05, 1.10999383951691684e-05)
```

Graph



Collins Aerospace

University of Minnesota