A DRONE'S EYE VIEW OF RUNTIME AND DESIGN-TIME AI IN SOFTWARE INTENSIVE SYSTEMS

RAISE @ ICSE 2019

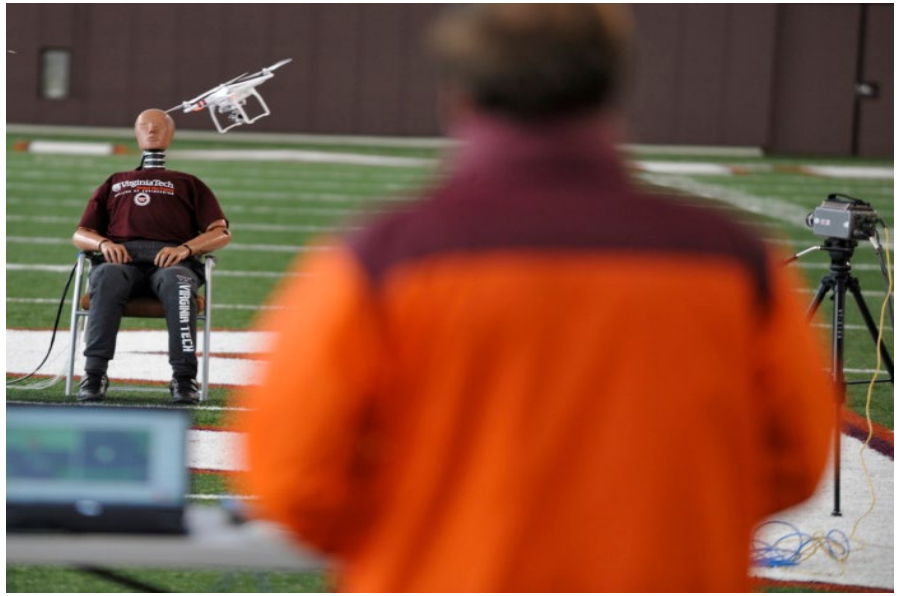
Jane Cleland-Huang, PhD
JaneHuang@nd.edu
Department of Computer Science and Engineering
University of Notre Dame

Much of the work described in this talk was funded by the US National Science Foundation under Grants CCF-0959924 and CCF-1265178.

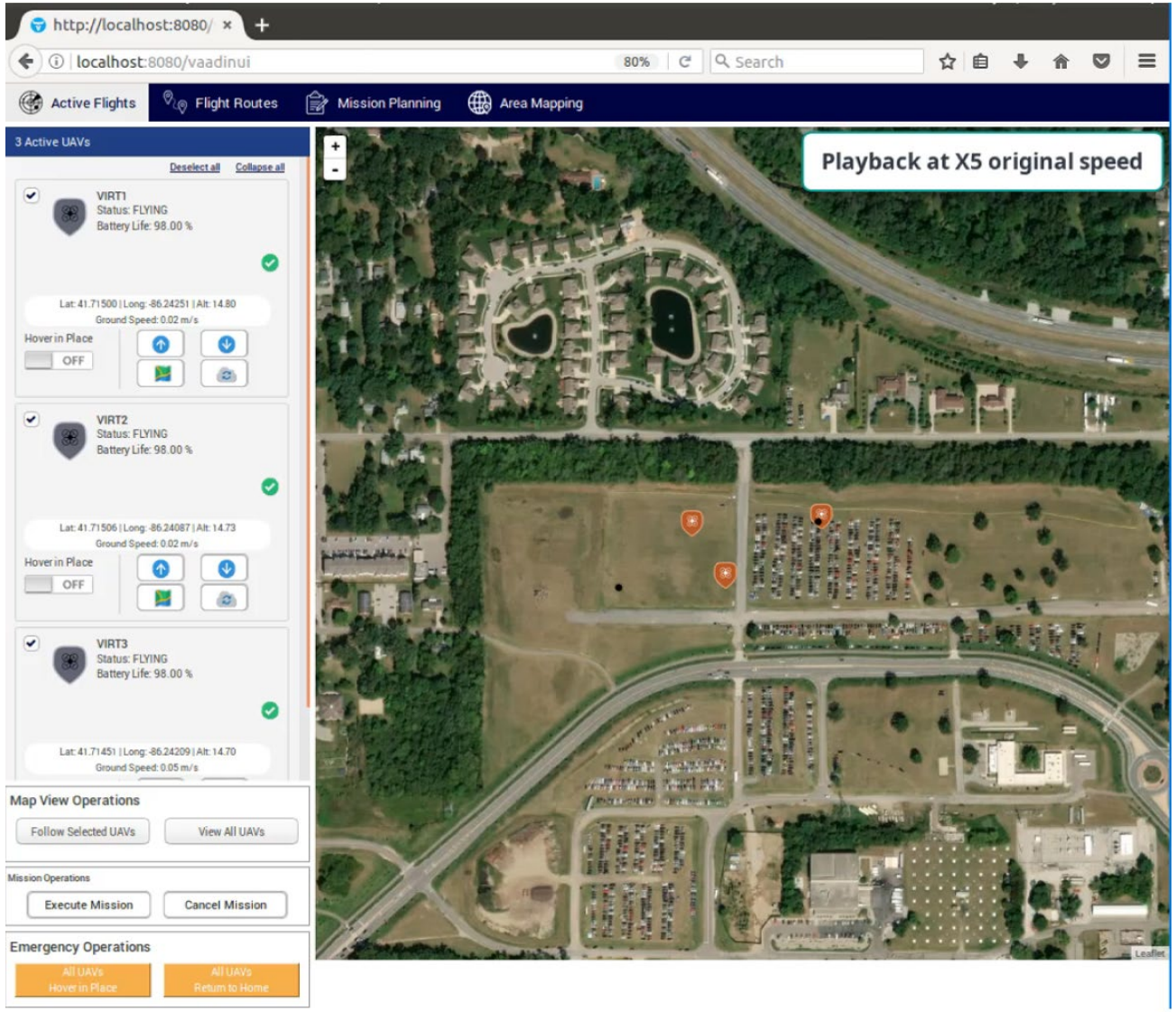
Drones and AI

May 16, 2015 - The audience reacts as a drone flies low over a crowd, now becoming commonplace at events.
(Jim Weber/The Commercial Appeal)

UAV deployments in urban contexts introduce safety concerns and provide a **case environment** for exploring safety-critical development practices such as the use of AI.



Dronology



Managing and controlling small Unmanned Aerial Systems (sUAS) in urban areas.

- Multi-drone mission control and planning
- Runtime monitoring & diagnostics
- Multi-drone simulation
- Runtime views
- Preflight checks

Drone Response

Structural Fire Surveillance

Structural Fire Rescue

http://sarec.nd.edu/pages/Dronology.html

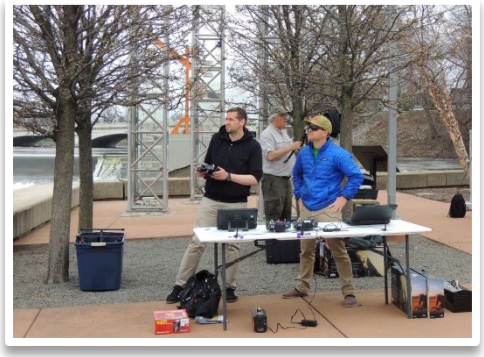
Hotspot detection on the roof, people detection through windows, fire-mapping.

AED Delivery

Drones delivering medical supplies must fly far beyond line of sight, circumvent obstacles and changing terrain, fly over urban areas, & deliver heavy payloads in potentially populous regions.

River Rescue Demo with Dronology

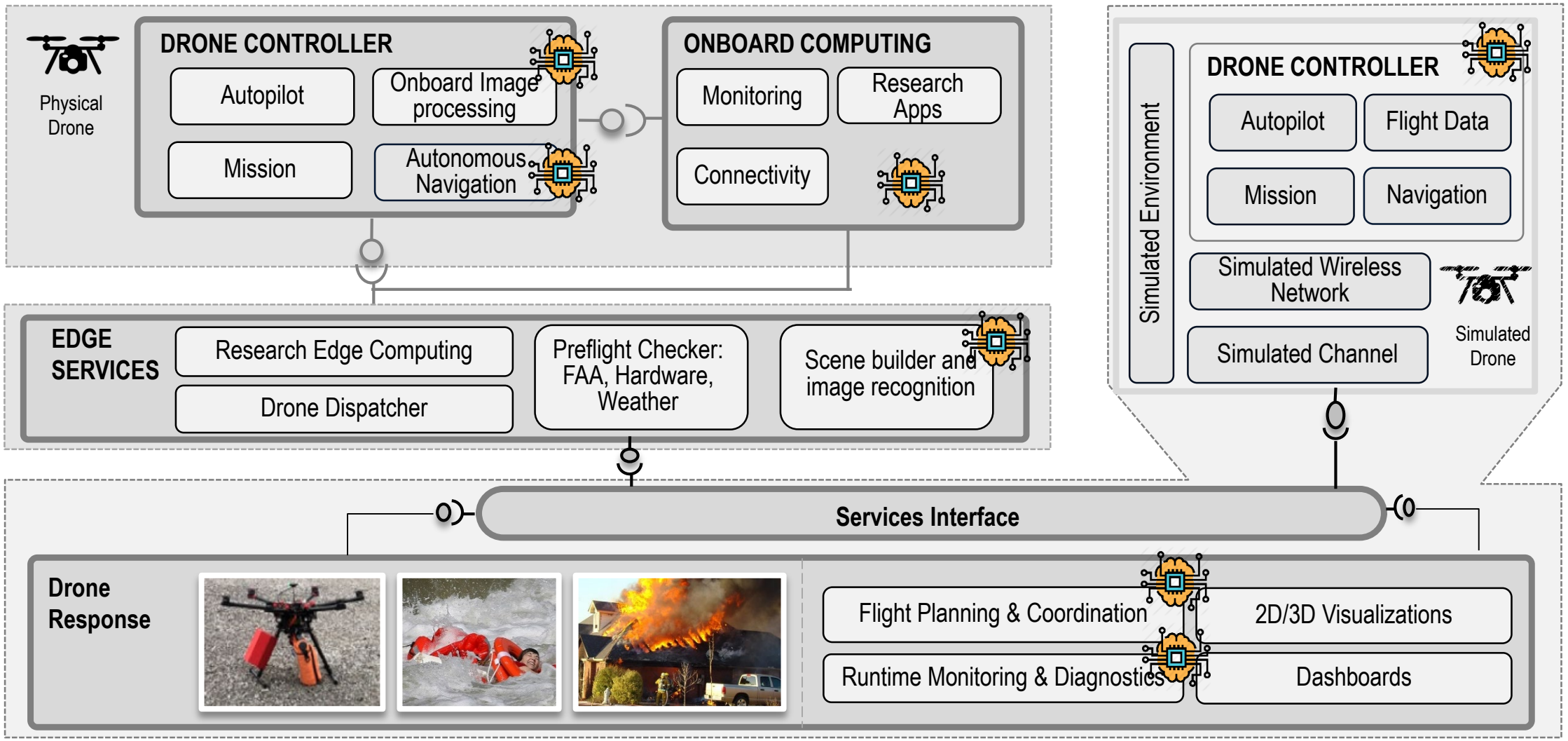
Deployment of drones for emergency response leverages AI while introducing safety concerns to address.



Drones as Tools

vs. Drones as Intelligent Partners

DroneResponse



A River Rescue Example

http://sarec.nd.edu/pages/Dronology.html

Mission Mode:
Searching

Rescuers:
1 boat
4 people

Drones:
3 Yuneec with thermal imagery

ALERTS

Drone 3 victim found

ACTIONS
Dispatch boat

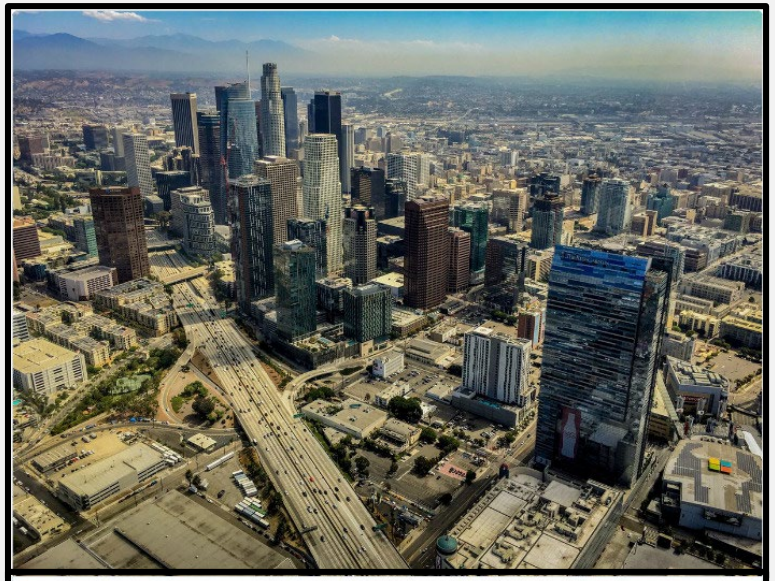
AI Components

- Route Planning
- Collision avoidance
- Object labeling and scene recognition
- Autonomous decision making
- Drone analytics

AI Example #1: A Drone's View of the World

City Describer
@CityDescriber
Follow

a traffic light hanging from a tree
11:42 AM - Sep 10, 2017
1 20 59



City Describer
@CityDescriber

a view of a cactus
3:42 AM - Sep 11, 2017
1 37 63

City Describer
@CityDescriber

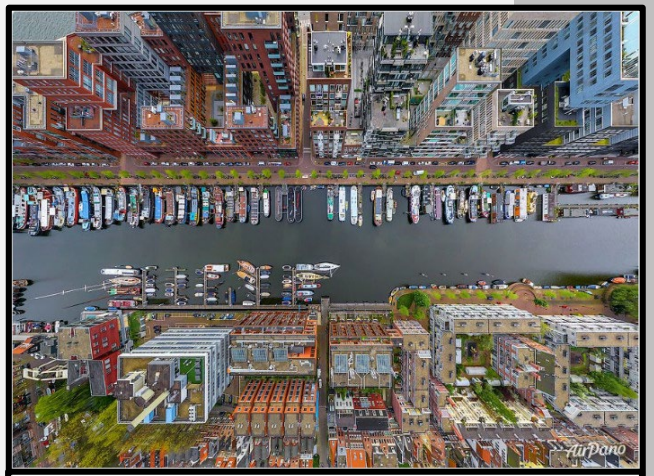
a close up of a tree
9:08 PM - Sep 9, 2017
4 9

City Describer
@CityDescriber
Follow

a table full of food
12:08 AM - Sep 14, 2017
2 32 77

City Describer
@CityDescriber
Follow

a close up of person riding a bike down a dirt road
11:39 PM - Sep 20, 2017
1 21 55



City Describer
@CityDescriber
Follow

a group of people in a city
8:55 PM - Sep 9, 2017
1 3

AI Example #1: Classification



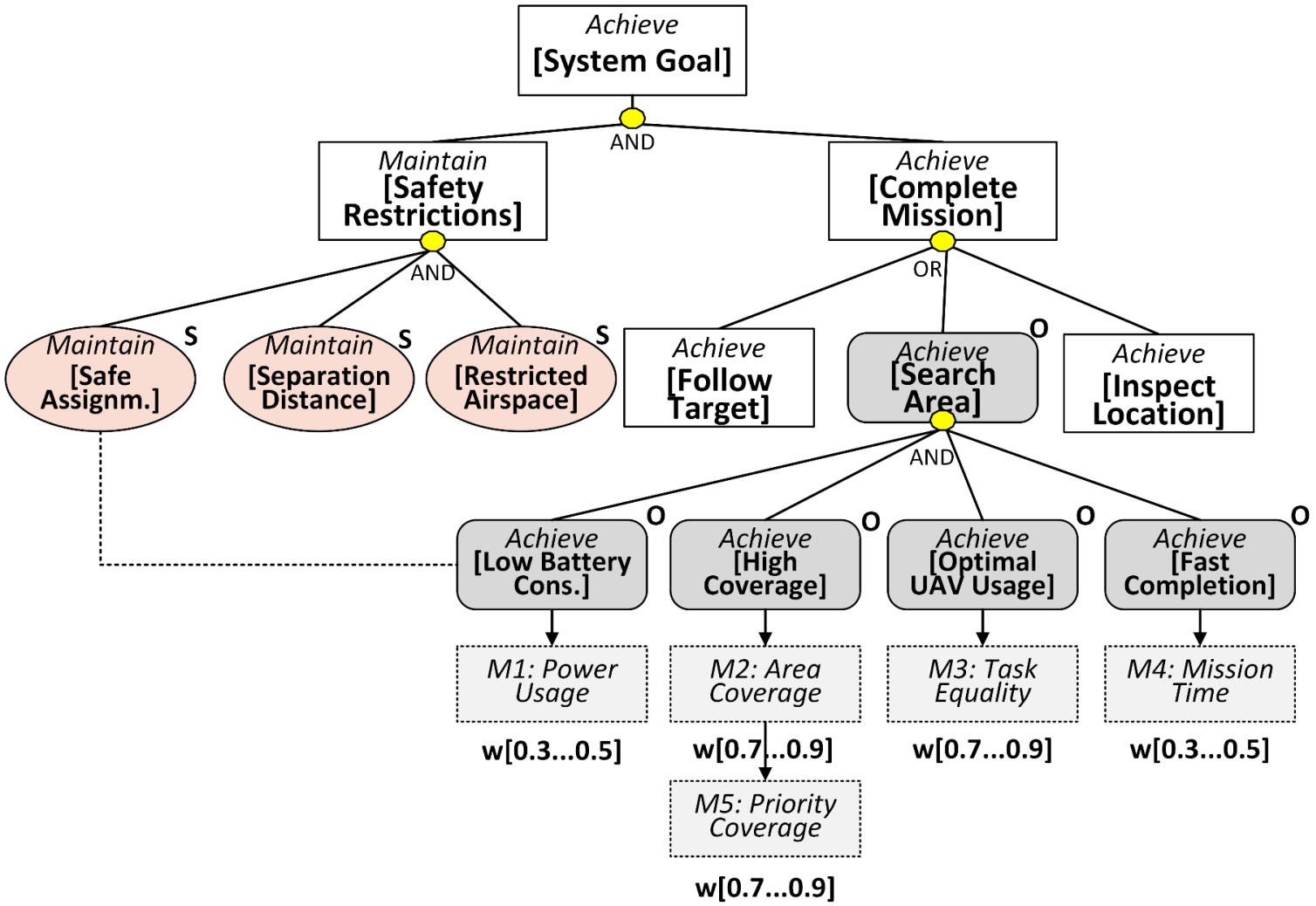






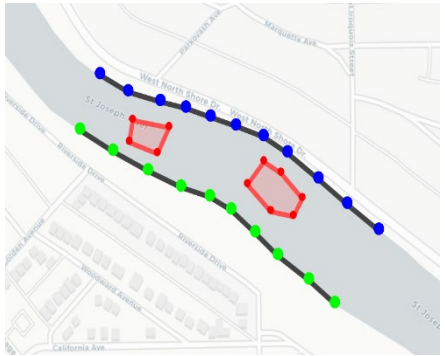


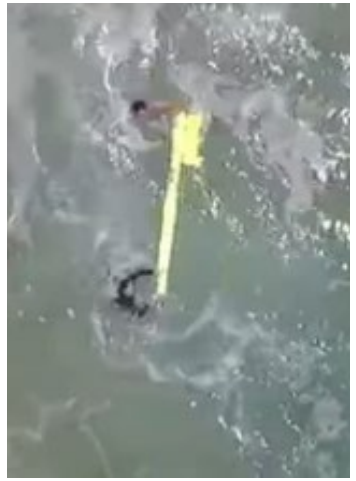

AI Example #2: Task Allocation



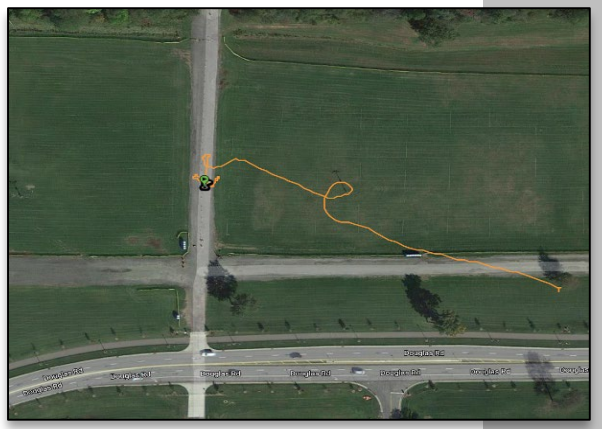
- Define goals for search & rescue.
- Incorporate both functional and safety goals.

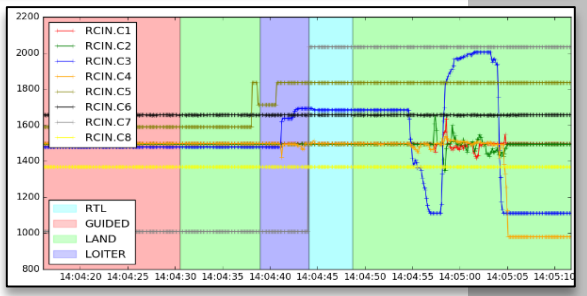
- Optimize system level goals through allocation of tasks to participating drones while meeting all safety and legal constraints.





AI Example #3: Onboard Analytics



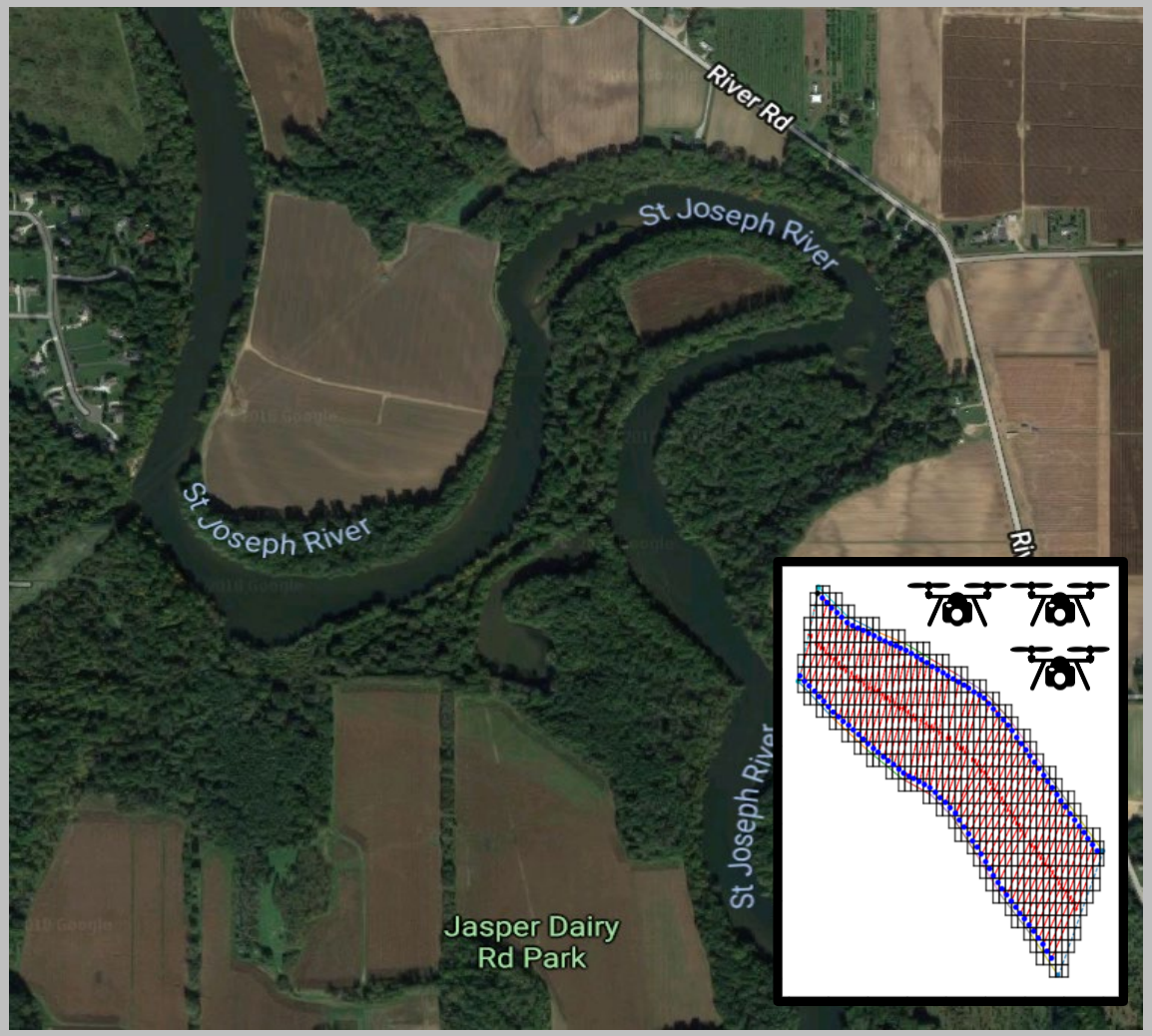


AI-driven collision avoidance and predictive health analytics.

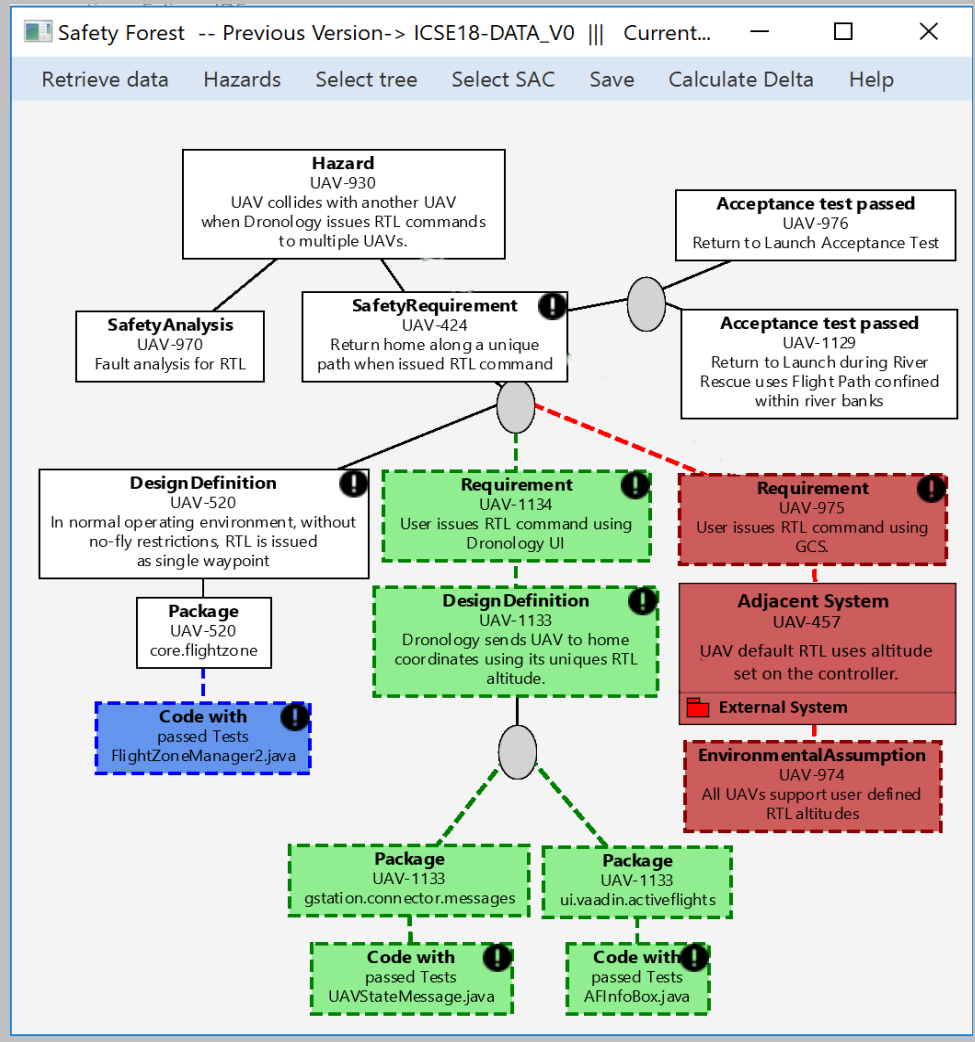
How should Software Engineers integrate AI components into Cyber-Physical Systems ?

AI at runtime and design time

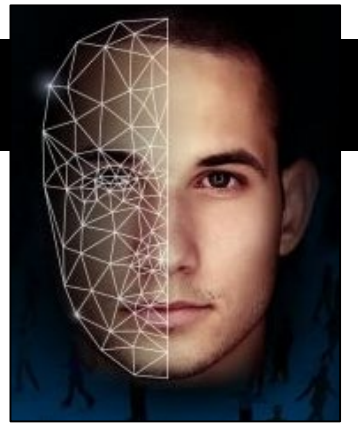
Runtime Decision Making



Static Safety Analysis



Framework for Trustworthy AI



Lawful AI

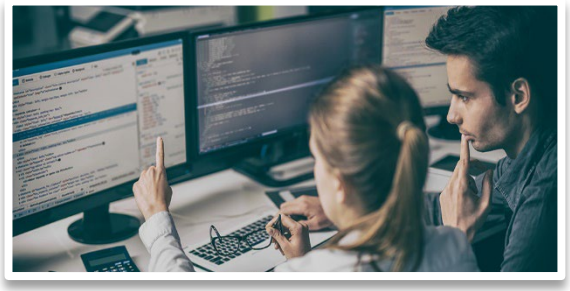
The law provides positive & negative obligations that describe what cannot be done, but also what should, or may, be done.

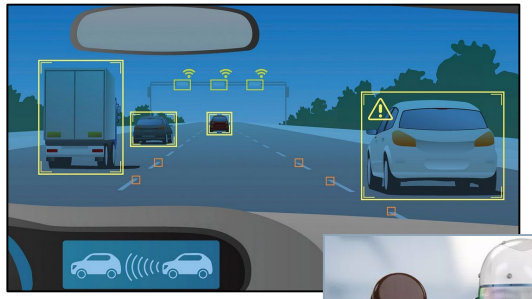
Ethical AI

Respect, serve and protect humans' physical and mental integrity, personal and cultural sense of identity, and satisfaction of their essential needs.

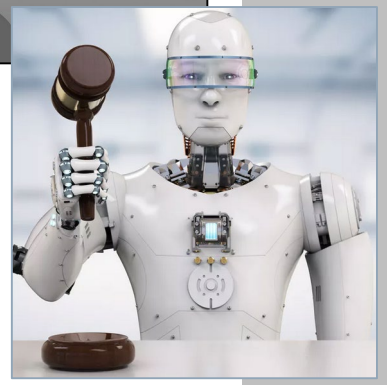
Robust AI

Perform in a safe, secure and reliable manner, with safeguards to prevent any foreseeable unintended adverse impacts.



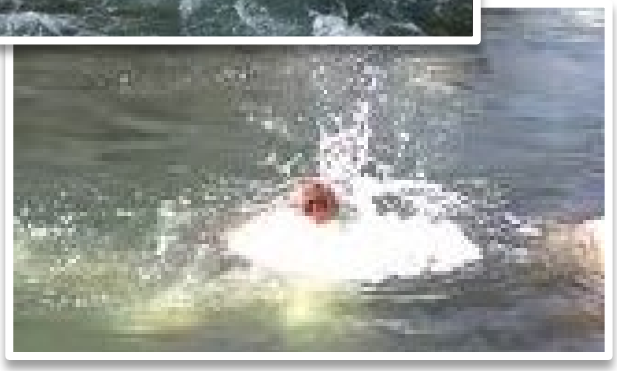


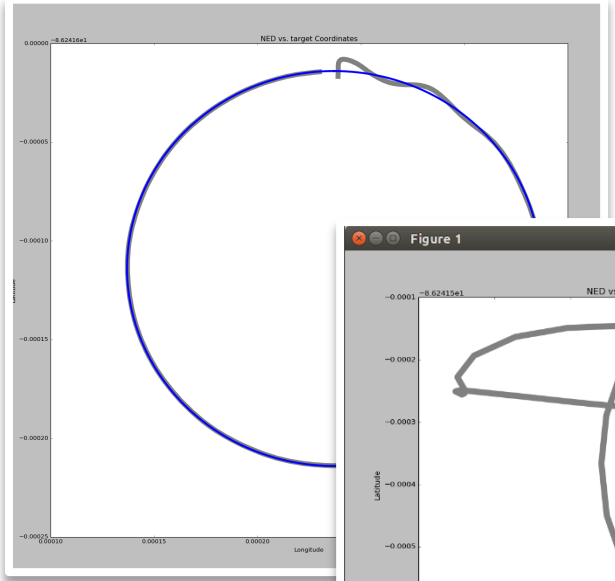
Software Engineers must support AI development, deployment and use to ensure that everyone can thrive in an AI-based world.

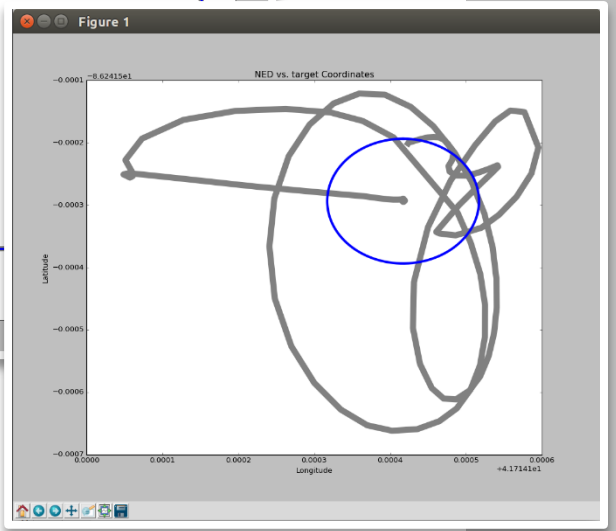


What does that mean for DroneResponse?

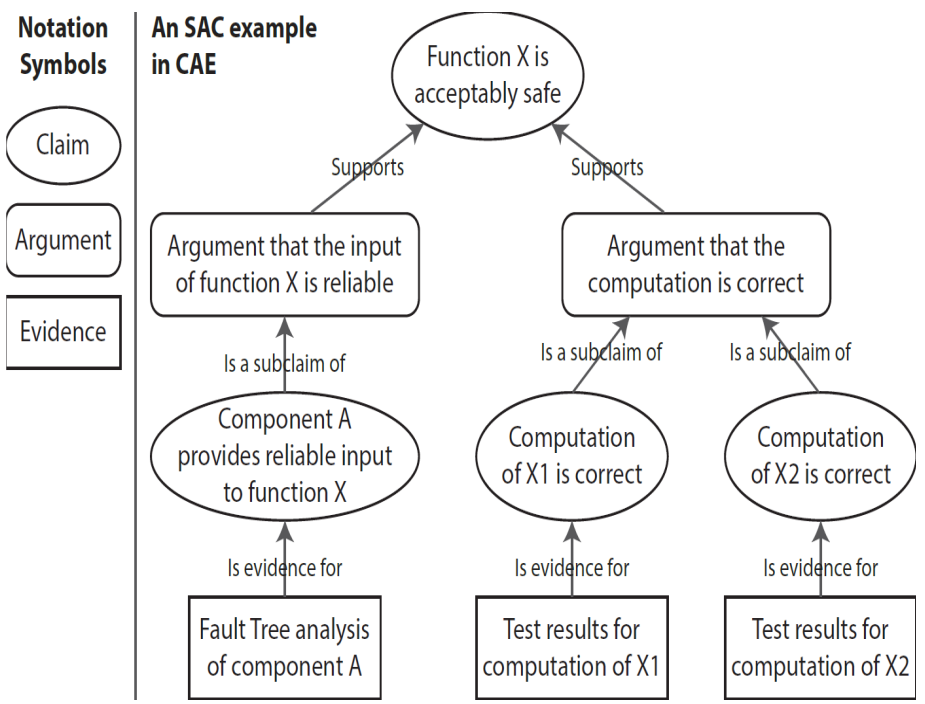
Lawful AI	Ethical AI	Robust AI
We follow FAA guidelines. We are trained and certified RPIC pilots. ✓✗	Our goal is to save people from drowning and burning. ✓✗	We follow sound Software Engineering principles with rigorous testing. ✓✗

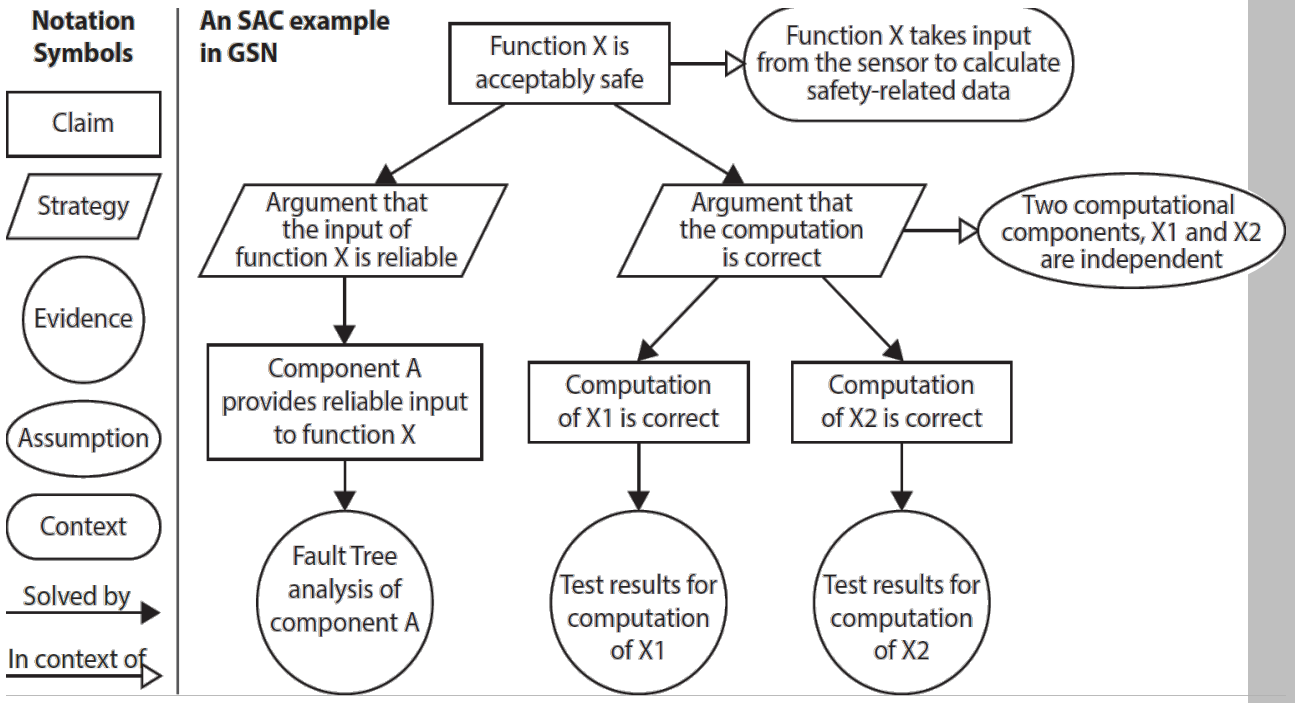






We need to build a robust Safety Case





Claim:
Assertion of compliance with key requirements and properties.
Must be within a specific context of use.

Arguments/Strategy:
Link evidence to claims via inference rules. Can be deterministic (true/false), probabilistic, or qualitative (i.e. link to regulations).

Evidence:
Process and people, testing, reviews, mathematical proofs.

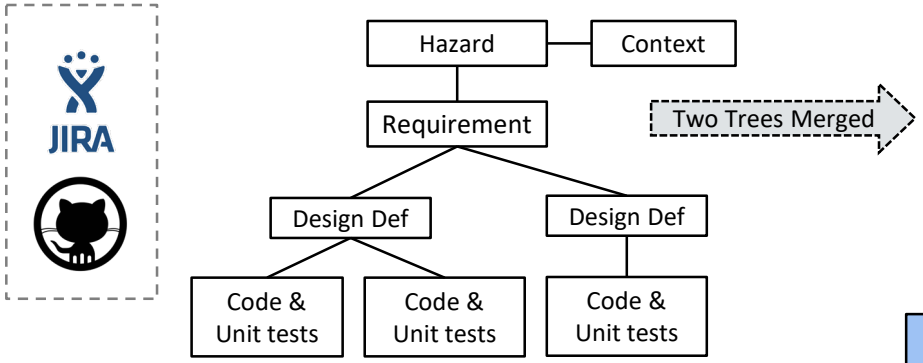
Context:

Environmental Assumptions

Examples provided by Jinghui Cheng

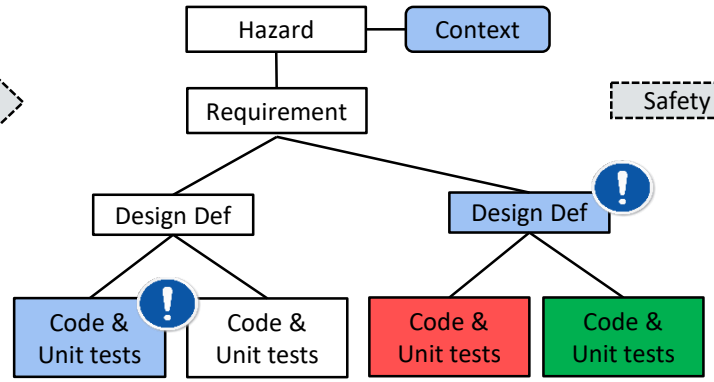
Our SAFA approach

Artifact View



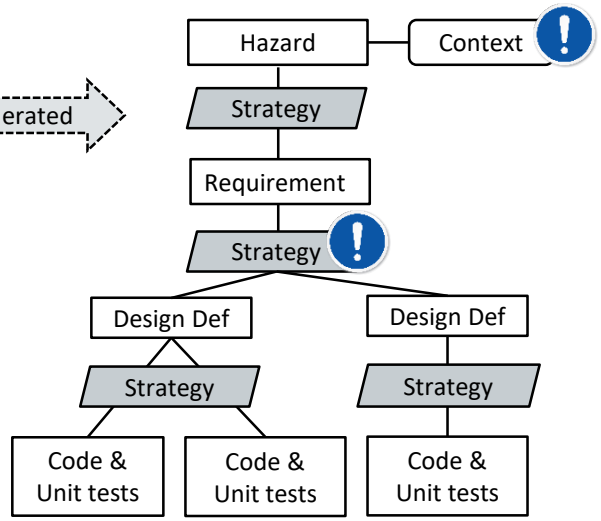
Project artifacts retrieved automatically from project repository and organized hierarchically according to traceability paths defined in the Traceability Information Model.

Delta View



The safety tree for the current version (V1) is automatically merged with the previous version (V0). Changes are depicted through colors(red=removed, green=added, blue=modified), and clickable information icons.

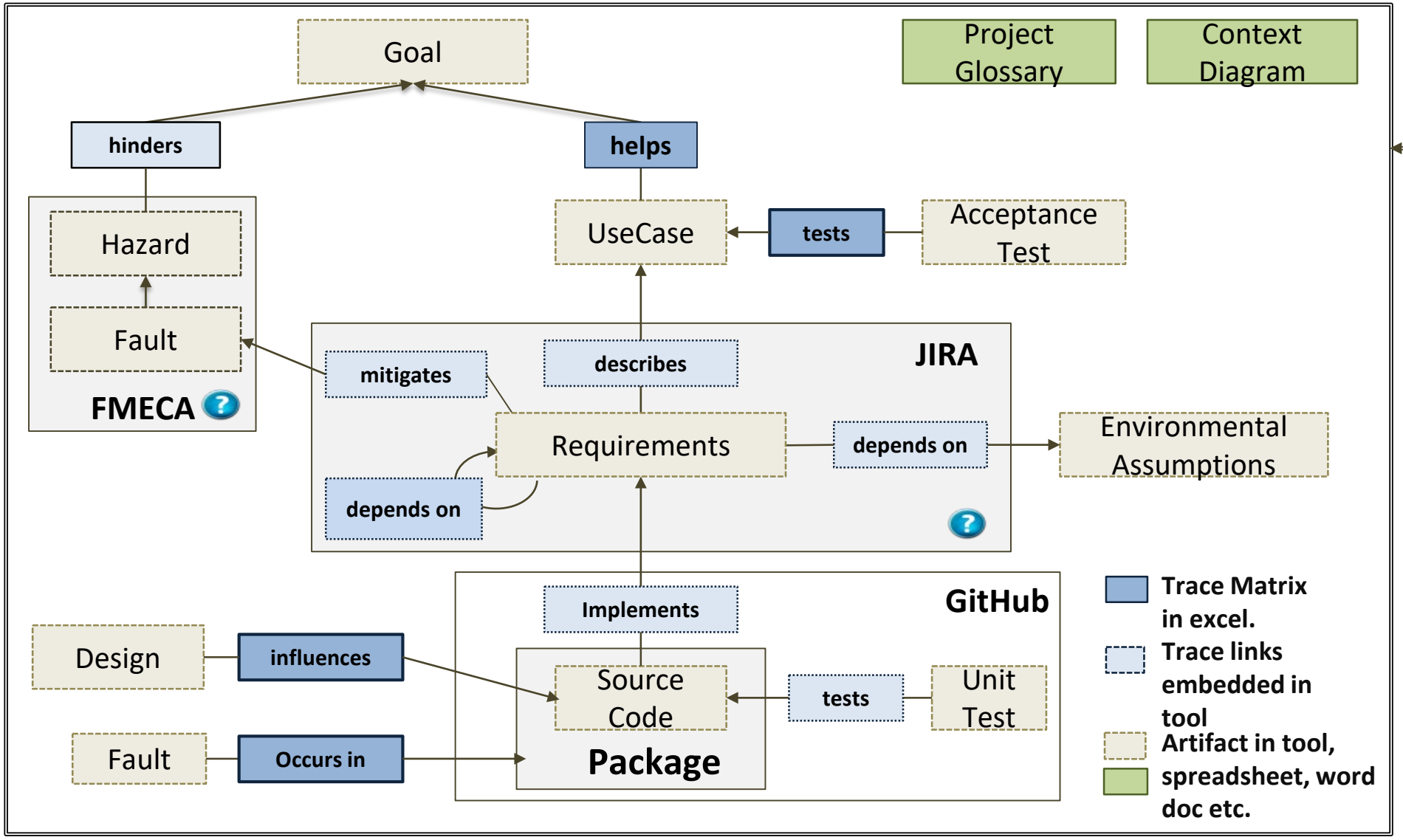
Safety View



The artifact tree augmented with strategy nodes and claims that provide a safety argument. Nodes are inserted automatically using heuristics.

An augmented FMECA could also be generated.

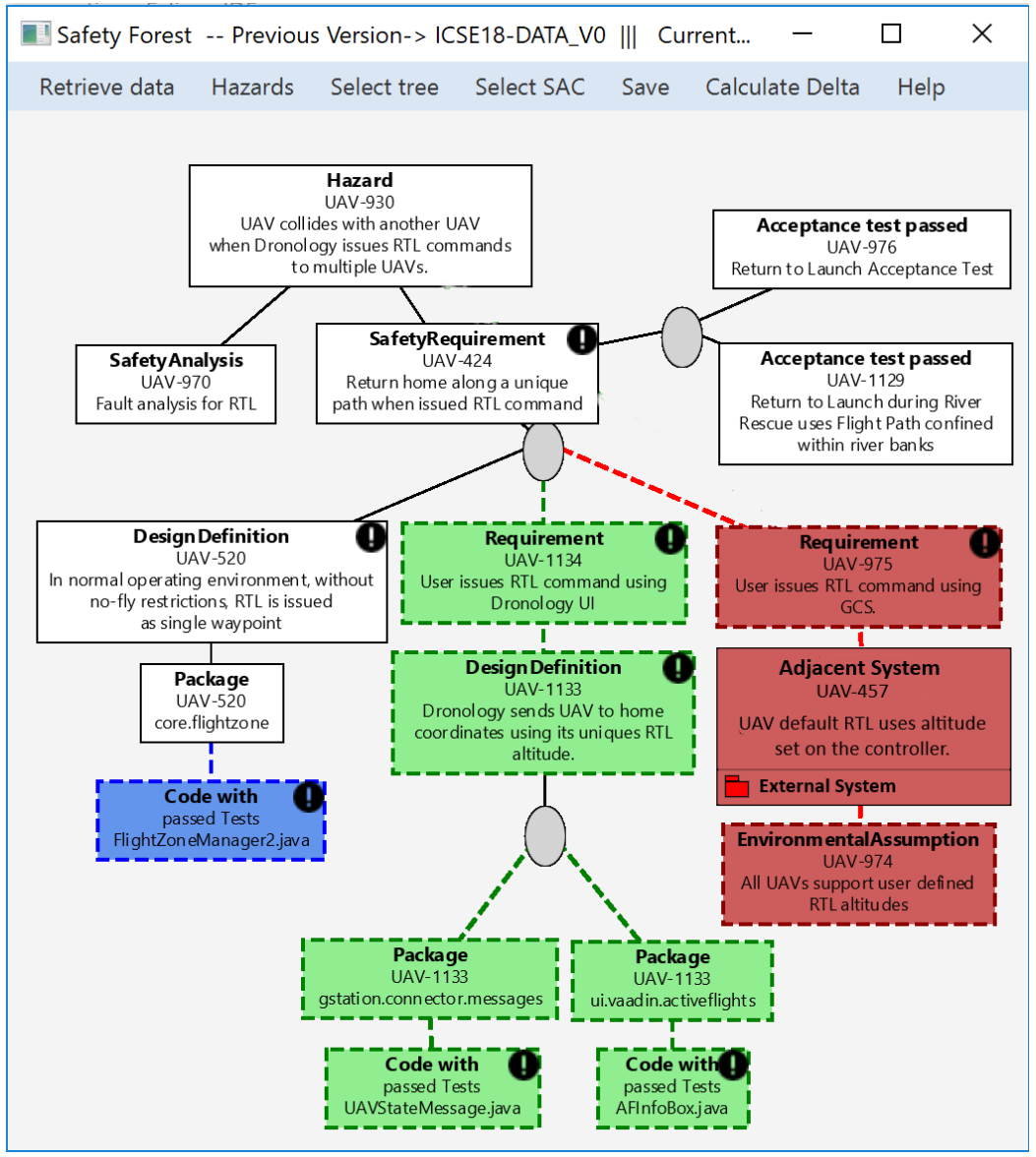
Be strategic



Three Golden Rules of Traceability

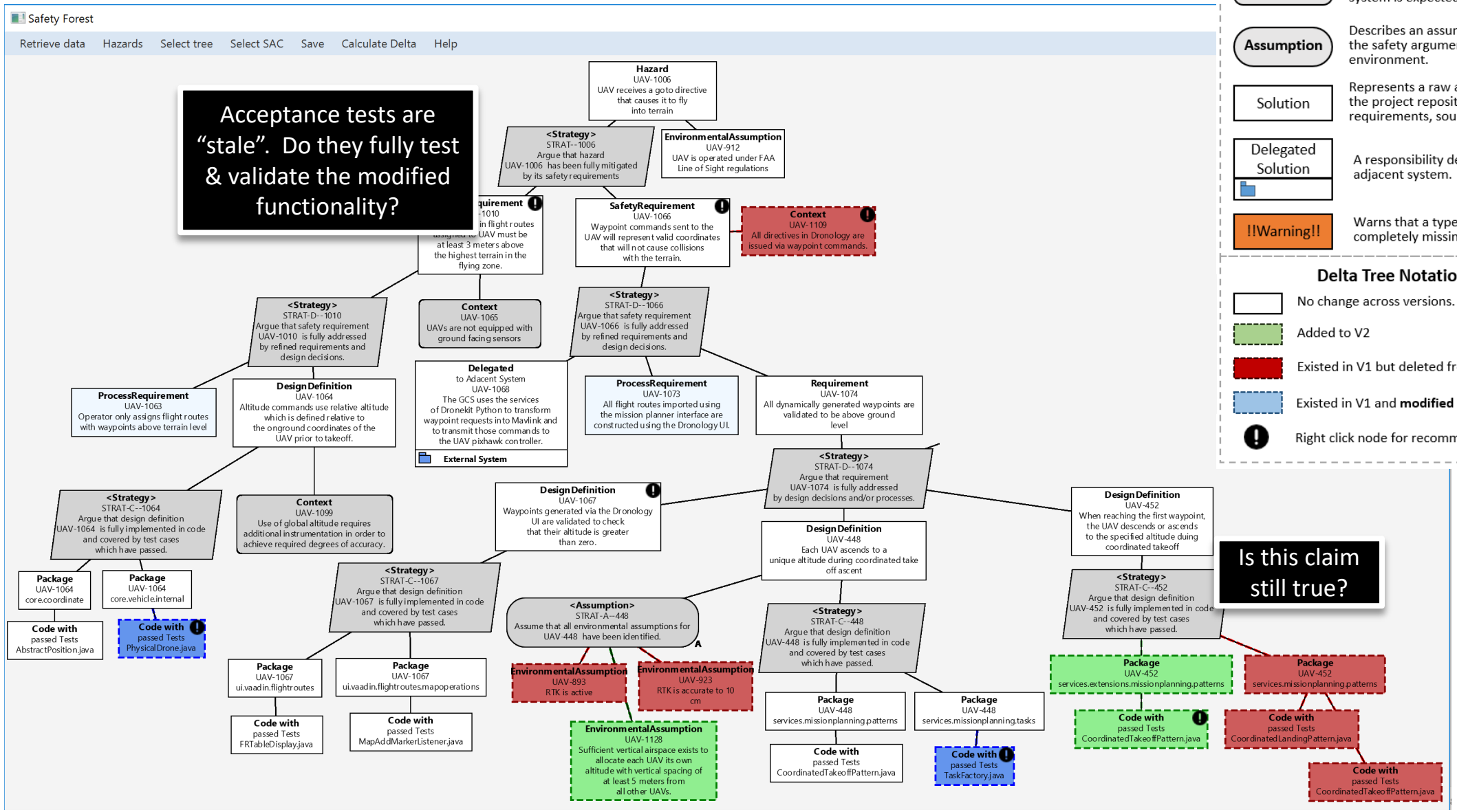
1. Trace purposefully
2. Get as much traceability for free as you can.
3. Automate where possible.

Delta View



- Depicts changes in the system that impact potential safety hazards.
- Designed to help analysts understand the change and to assess its impact upon system safety.
- Informational nodes provide information about the change and actionable recommendations.

Transformation to a Safety Case



Proof of Concept User Study:

To what extent does SAFA support an analyst in identifying safety impacts?
Two treatments. T1: View artifact trees for v1 and v2, T2: View delta tree

Version 1 (v1)
49,400 LOC
418 Java Classes
146 Requirements
224 Design definitions

Version 2 (v2)
73,591 LOC
646 Java classes
185 Requirements
283 Design definitions

ID	Role	Domain	Yrs	SC
P1	Software Engineer	Aviation & Defense	8	Y
P2	Developer	Operating Systems	1	N
P3	Developer	Development	2+	Y
P4	Developer	Embedded Systems	1	N
P5	Developer	Embedded Systems	2	Y
P6	Software Engineer	Software Development	7	Y
P7	Developer	Information Systems	2	N
P8	Software Engineer	Unmanned Aerial Systems	1	Y
P9	Systems Engineer	Embedded Systems	35	Y
P10	Requirements Engineer	Defense Systems	23	Y

Q	Theme	Description	Cnt
Q3	Visualization	The extent to which color coding and other visualizations highlight issues	6
Q3	Speed	The extent to which problems can be identified quickly	6
Q3	Informative	The extent to which the provided information supports safety analysis	5
Q3	Process	The ease of the analysis process	5
Q3	Trust	The trust that a user places in SAFA to identify problems	1
Q4	Code insight	The ability of SAFA to identify and display impactful code changes	3
Q4	Rationale	Rationales explaining additions, deletions, or modifications of artifacts.	2

I would kill to have SAFA in my workplace

Give me back the delta view!

I find myself implicitly trusting the tool. Is the tool certified?

Back to our example

Mission Mode:
Searching

Rescuers:
1 boat
4 people

Drones:
3 Yuneec with thermal imagery

ALERTS

Drone 3 victim found

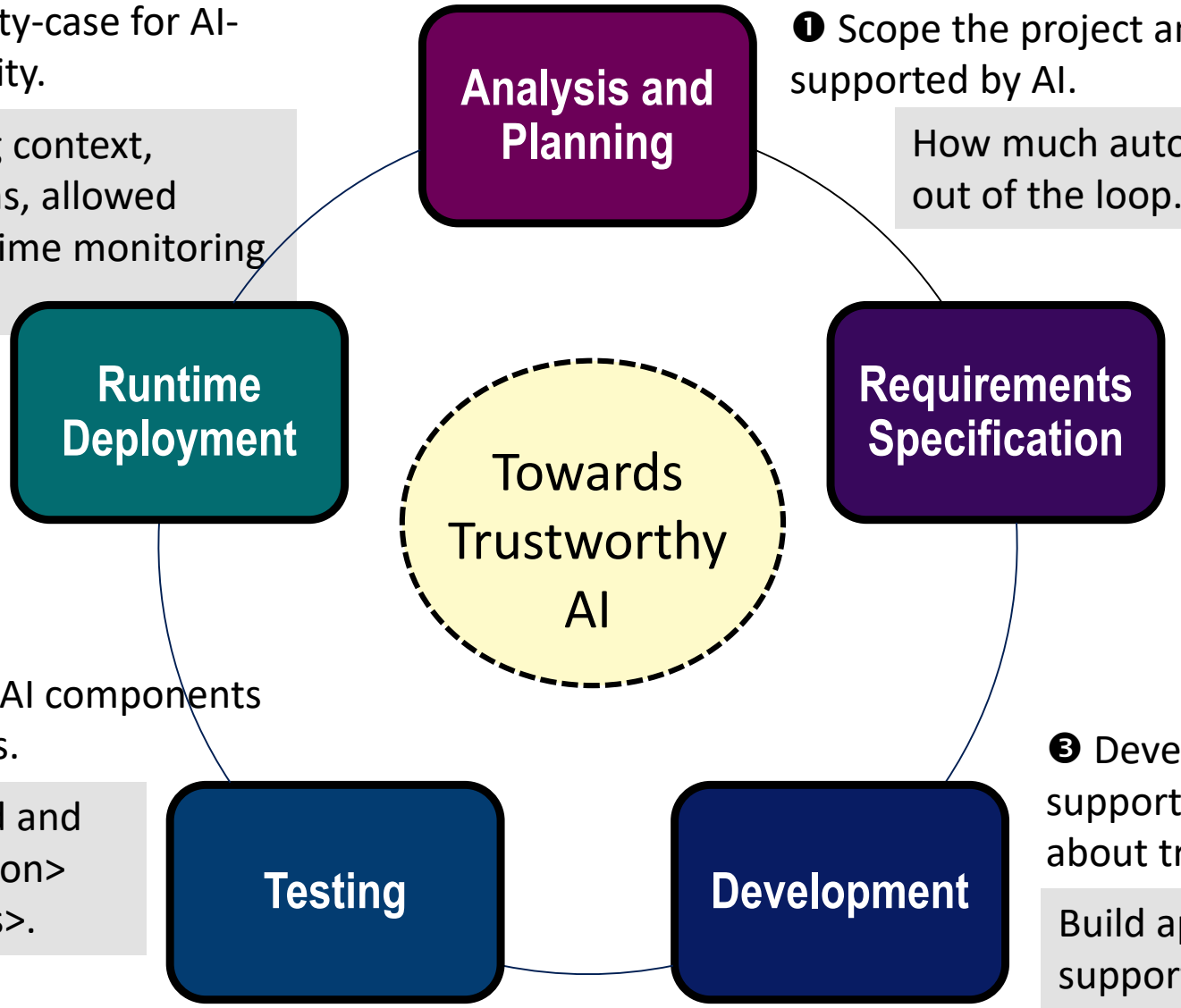
ACTIONS
Dispatch boat

AI: Victim detection

- Under what circumstances was the victim-detection image recognition trained?
- Does the current context match that context?
- What certainty is the decision made with?
- What supporting actions are needed?

Landing a Drone on a moving object

How does AI impact the life-cycle?



❶ Scope the project and identify parts supported by AI.

How much autonomy? Human in or out of the loop.

❷ AI specific requirements

The <AI Component> shall <action> under <conditions>

❸ Develop and train AI components and supporting infrastructure. Be strategic about training data.

Build appropriate training sets for supporting <action> under <conditions>

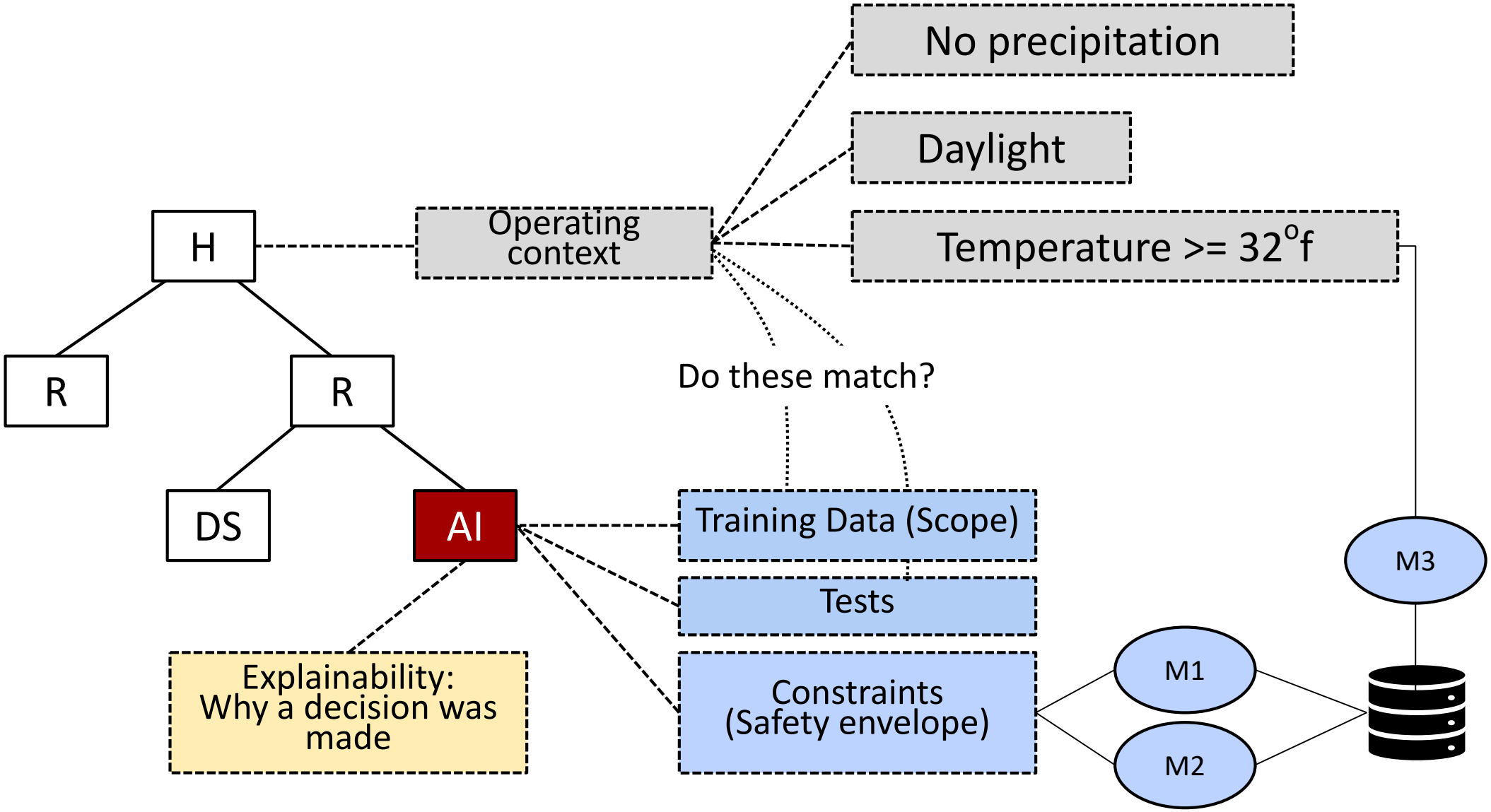
❹ Rigorously test AI components in diverse contexts.

Tests are designed and executed for <action> under <conditions>.

❺ Establish a safety-case for AI-related functionality.

Integrate training context, current conditions, allowed actions with runtime monitoring into safety case.

Making a Safety Case for AI components

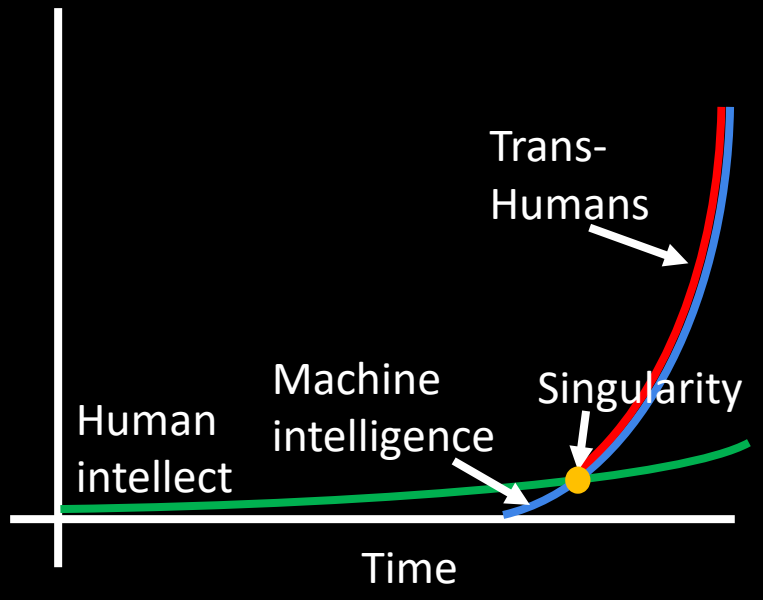


Let an ultra-intelligent machine be defined as a machine that can **far surpass all the intellectual activities of any man** however clever.

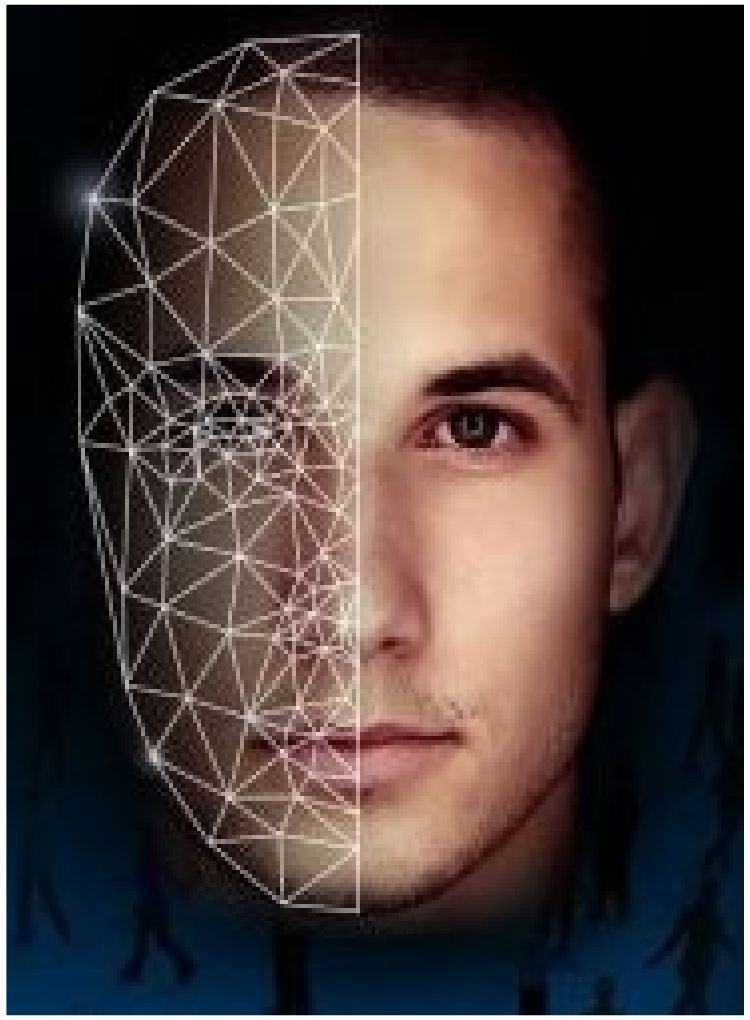
Since the design of machines is one of these intellectual activities, **an ultra-intelligent machine could design even better machines**; there would then unquestionably be an "intelligence explosion," and the intelligence of man would be left far behind.

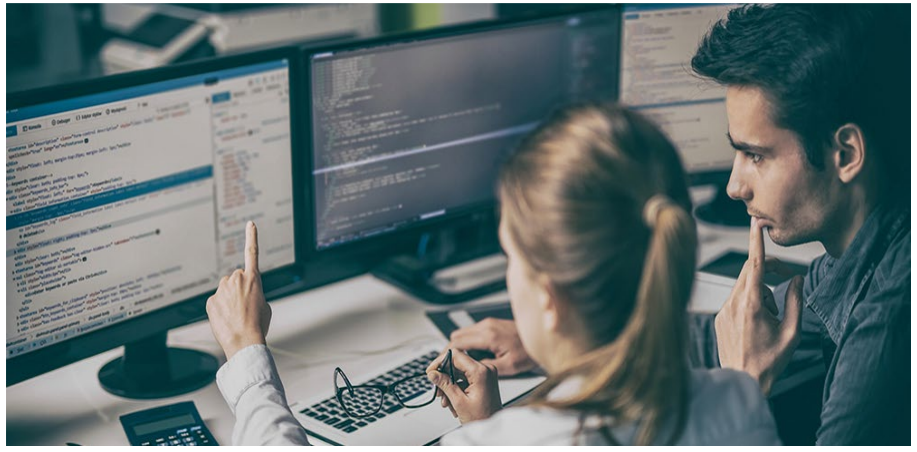
Thus the first ultra-intelligent machine is the last invention that man need ever make . . .

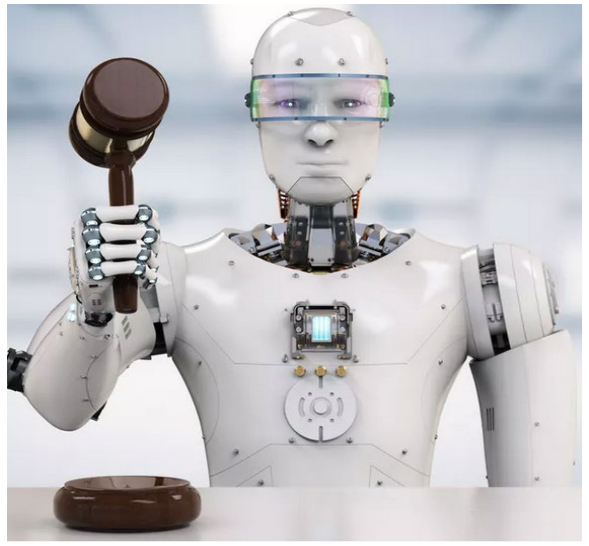
I.J.Good: Chief statistician for Alan Turing's code-breaking team in World War 2.

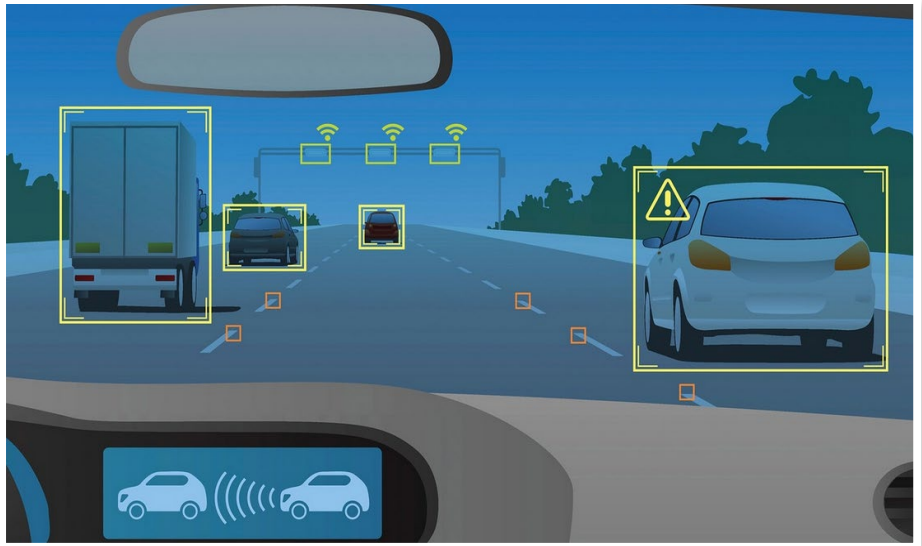


The SE Charge









As Software Engineers we need to **discover**, **ask**, **answer**, and **act** upon the hard questions associated with implementing AI in practical applications that impact society.

A DRONE'S EYE VIEW OF RUNTIME AND DESIGN-TIME AI IN SOFTWARE INTENSIVE SYSTEMS

RAISE @ ICSE 2019

Jane Cleland-Huang, PhD
JaneHuang@nd.edu
Department of Computer Science and Engineering
University of Notre Dame

Some of the work described in this talk was funded by the US National Science Foundation under Grants CCF-0959924 and CCF-1265178.