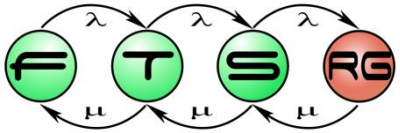
Towards using ML in critical applications

András Pataricza
Budapest University of Technology and Economics
Dept. Measurement and information Systems
With contributions of
István Majzik, Zoltán Micskei, András Vörös and András Földvári

The research reported in this presentation was supported by the BME-Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI)

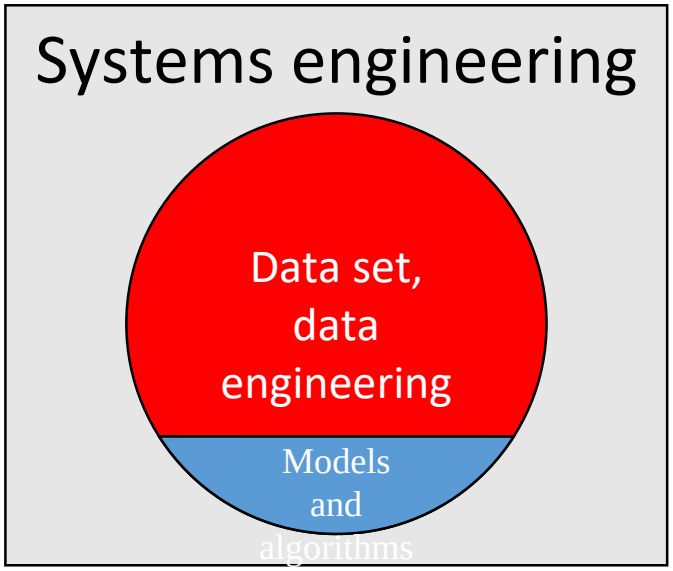


Focal problem

- AI solutions are highly effective
- BUT:
  can we trust them?
- Black box?
- What are the impacts
  - Integrity of the decision
  - Representativeness of the teaching set
  - Coverage

IN CRITICAL APPLICATIONS

Challenges



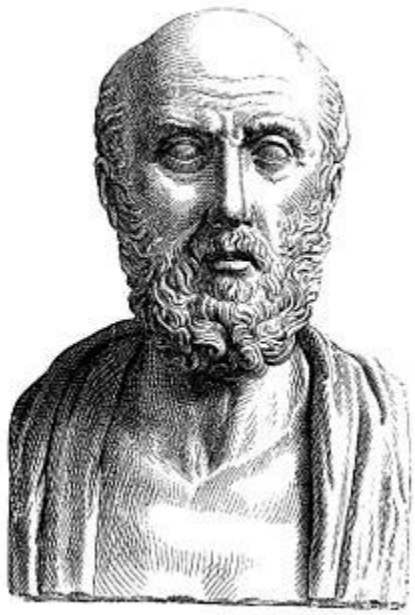
Source: Andrej Karpathy: Building the Software 2.0 Stack

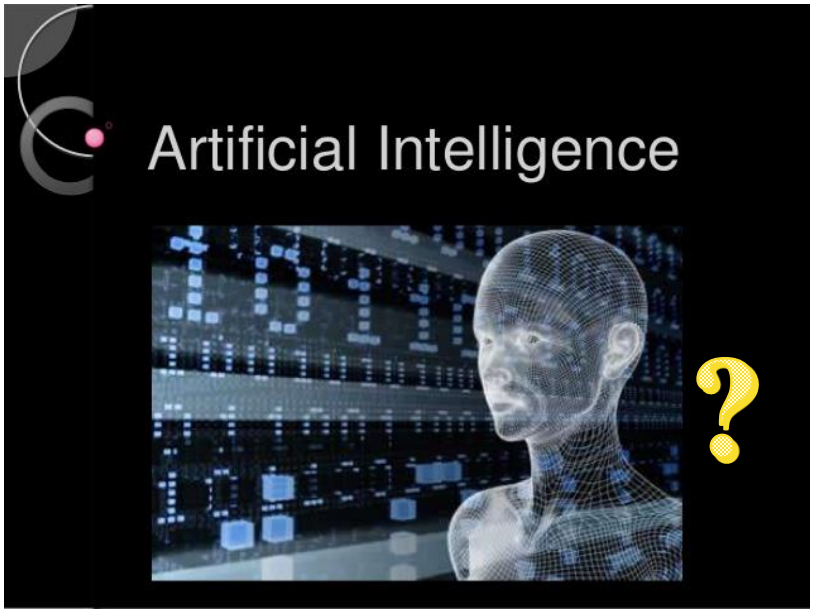
- Dataset:
  - Teaching
  - Testing
- Debugging:
  - Interpretability/Explainability
    - Using a 90% accurate model we understand, or
    - 99% accurate model we don't.
- Faults/attacks, fault-tolerance/defenses
- Safety
- Security

Hippocratic Oath

I swear by Apollo the Healer, by Asclepius, by Hygieia, by Panacea, and by all the gods and goddesses, making them my witnesses, that I will carry out, according to my ability and judgment, this oath and this indenture.

- **To hold my teacher in this art equal to my own parents;**
- I will use treatment to help the sick according to my ability and judgment, but **never with a view to injury and wrong-doing**





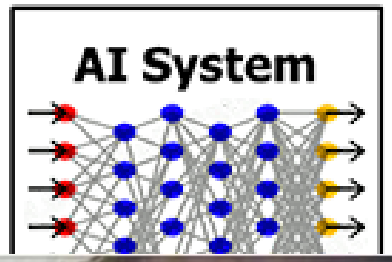
https://image.slidesharecdn.com/a-i-presentation-121229232307-phpapp02/95/artificial-intelligence-1-638.jpg?cb=1356823463

Explainable Artificial Intelligence (XAI)

DARPA-BAA-16-53

Mr. David Gunning : https://www.darpa.mil/program/explainable-artificial-intelligence

The Need for Explainable AI

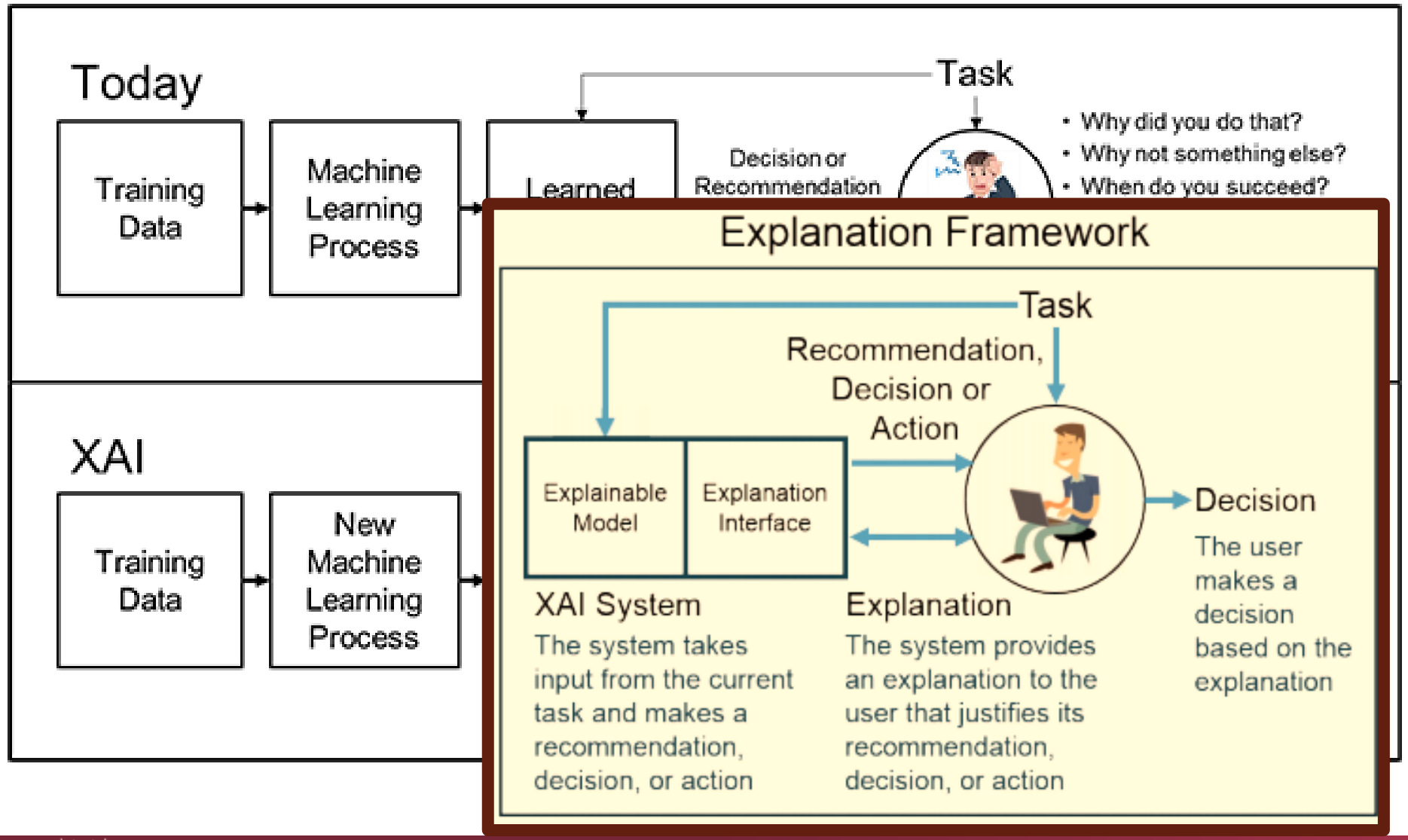


DoD and non-DoD Applications

Transportation

- Why did you do that?
- Why not something else?
- When do you succeed?
- When do you fail?
- When can I trust you?
- How do I correct an error?

XAI Concept

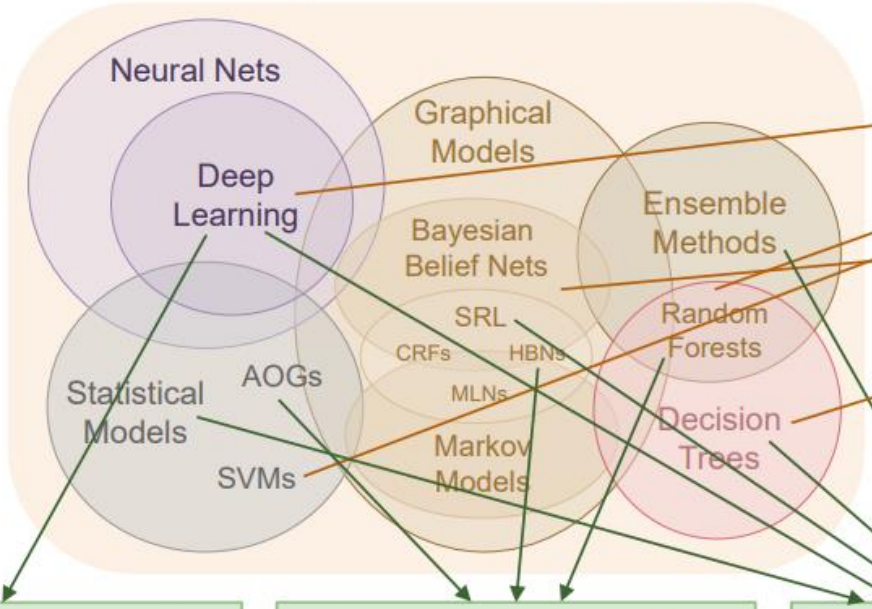

Today
Training Data
Machine Learning Process
Learned
Task
Decision or Recommendation
• Why did you do that?
• Why not something else?
• When do you succeed?
XAI
Training Data
New Machine Learning Process
Explanation Framework
Task
Recommendation, Decision or Action
Explainable Model
Explanation Interface
Decision
The user makes a decision based on the explanation
XAI System
The system takes input from the current task and makes a recommendation, decision, or action
Explanation
The system provides an explanation to the user that justifies its recommendation, decision, or action

Performance vs. Explainability

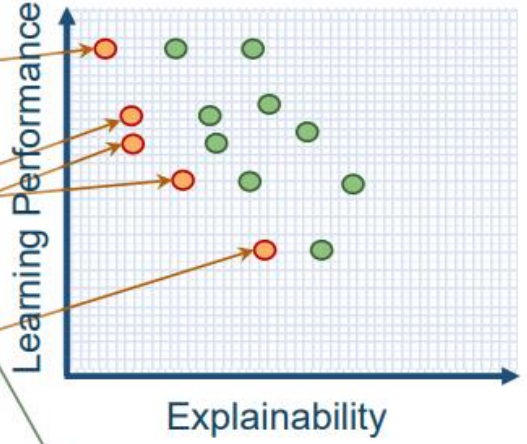
New Approach

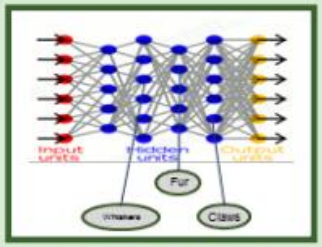
Create a suite of machine learning techniques that produce more explainable models, while maintaining a high level of learning performance

Learning Techniques (today)



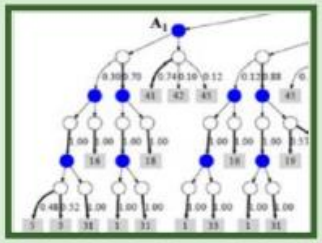
Explainability (notional)





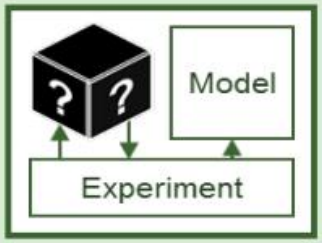
Deep Explanation

Modified deep learning techniques to learn explainable features



Interpretable Models

Techniques to learn more structured, interpretable, causal models



Model Induction

Techniques to infer an explainable model from any model as a black box

Requirements

Cyber-Physical Systems

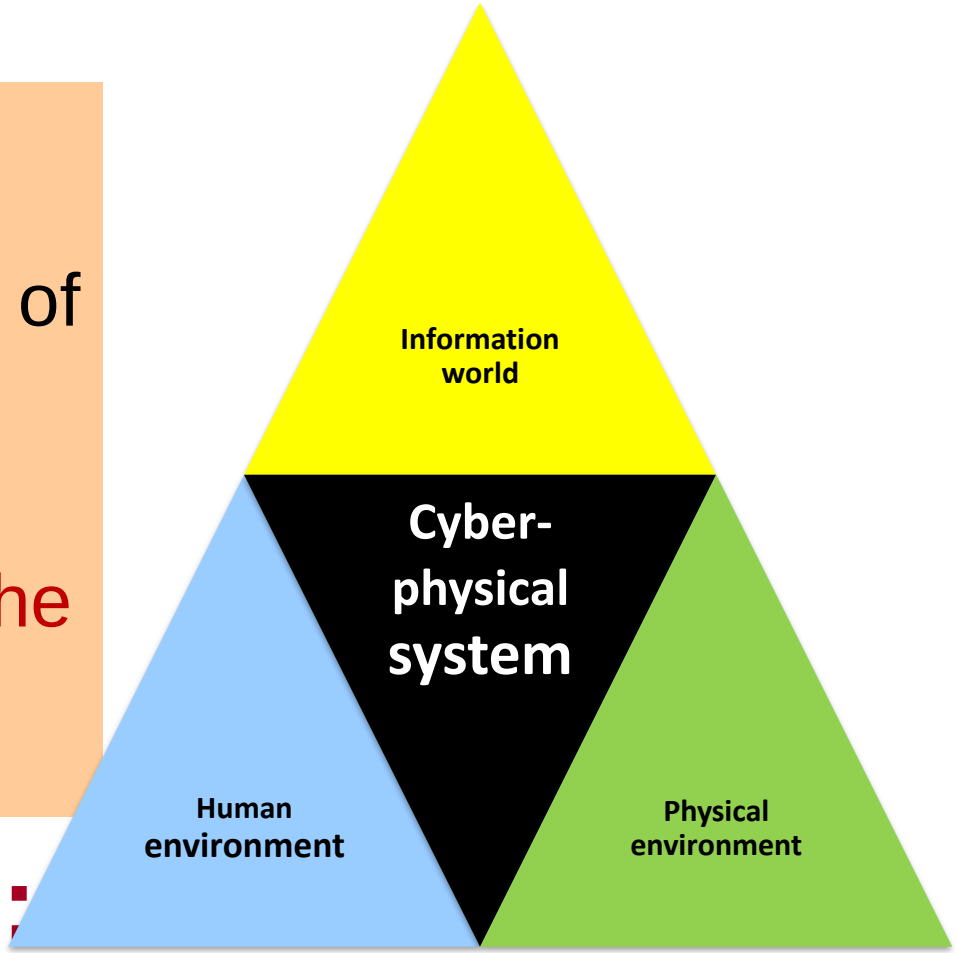
Cyber-Physical Systems definition

- *“Cyber-Physical Systems or "smart" systems are co-engineered interacting networks of physical and computational components. These systems will provide the foundation of our critical infrastructure, form the basis of emerging and future smart services, and improve our quality of life in many areas.”*

National Institute of Standards and Technology
U.S. Department of Commerce

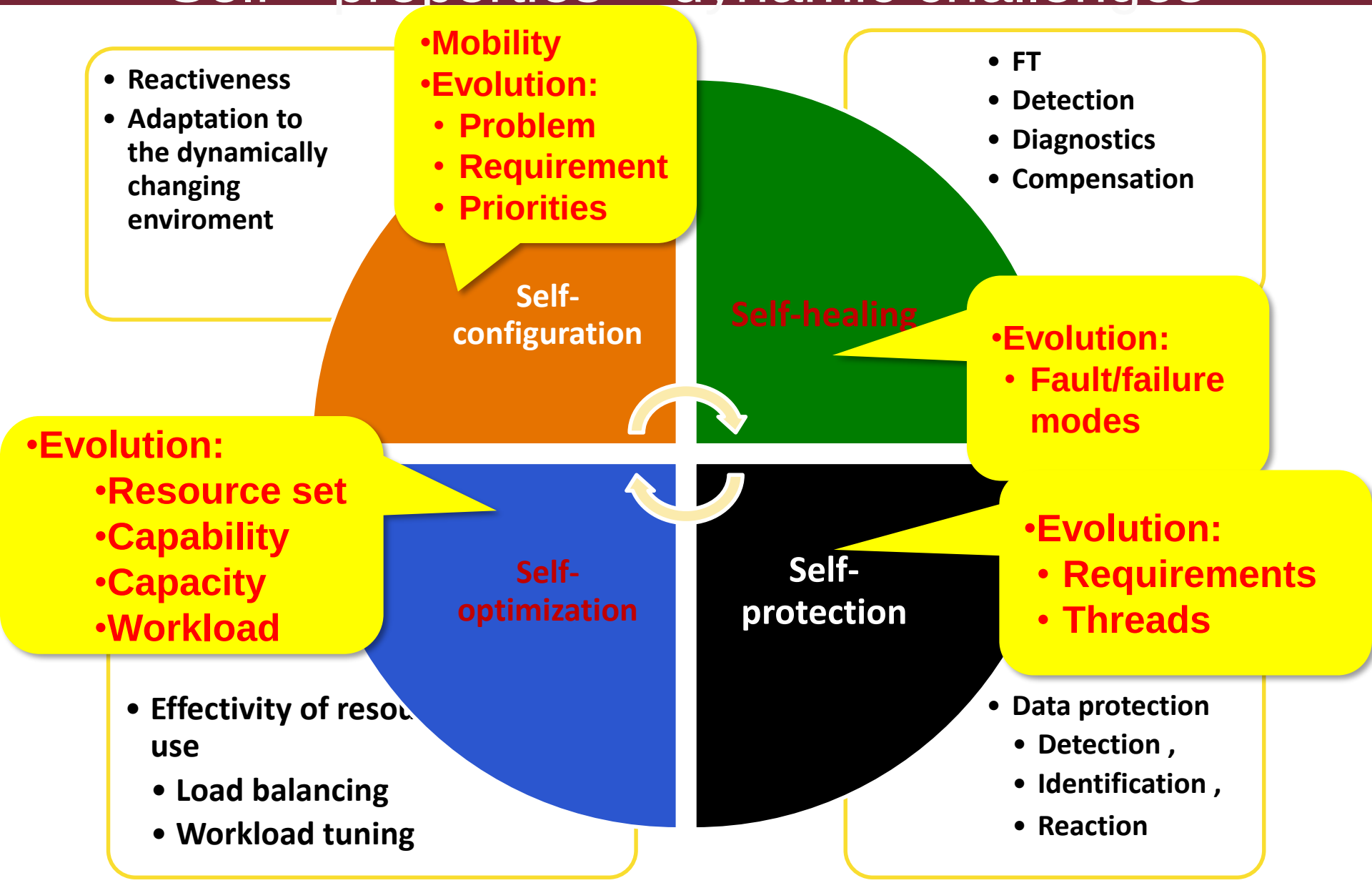
Let's reach an unlimited intelligence by the synergy of

- intelligence in the cyber space and
- ES interfacing them to the physical world



THE NEW ERA: INTERNET OF THINGS AKA CYBER-PHYSICAL SYSTEMS

Self-* properties – dynamic challenges

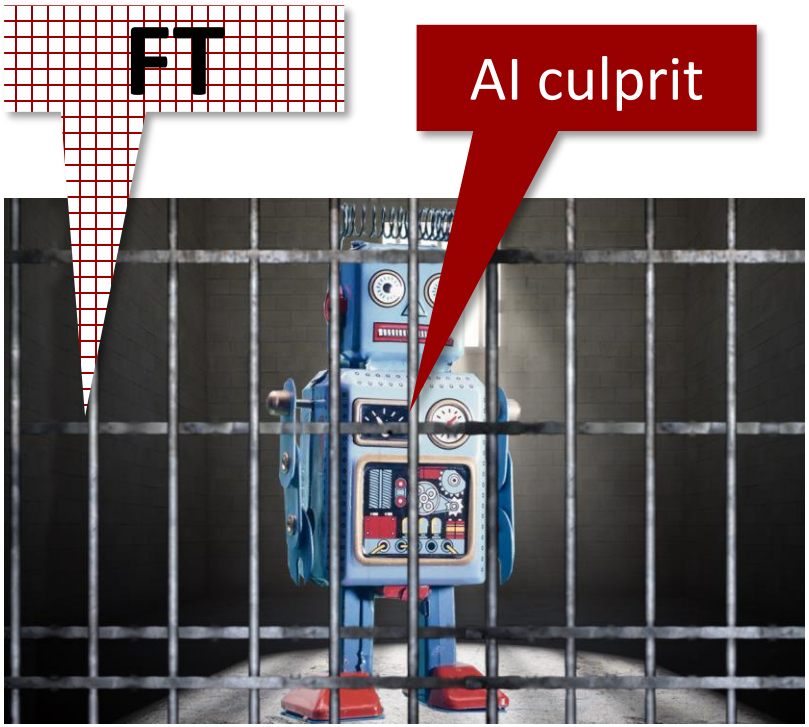


Critical applications

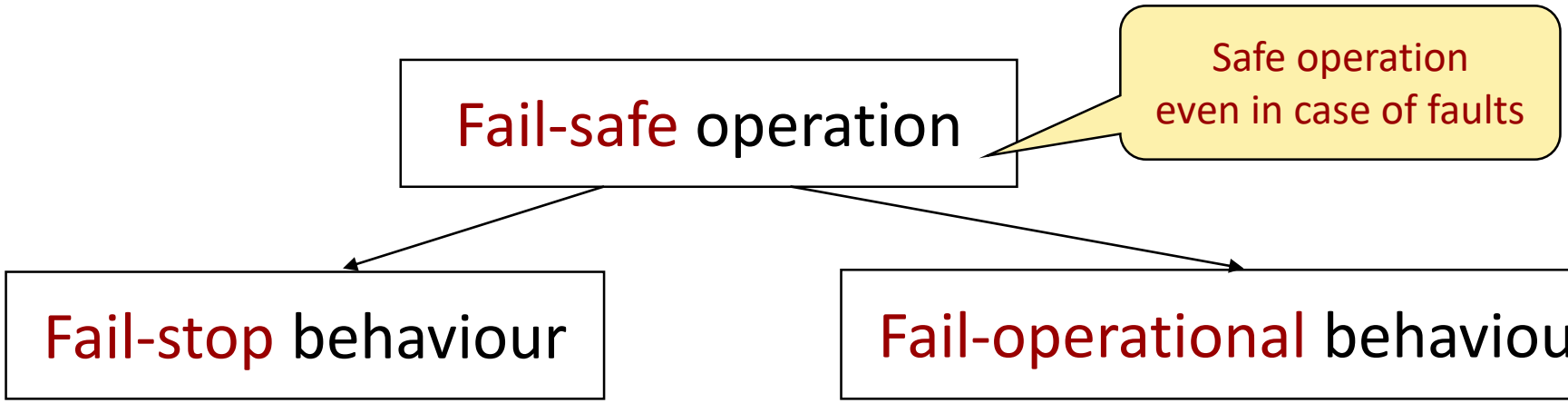
- What we specified?
  - Safety function requirements: Function which is intended to achieve or maintain a safe state
  - Safety integrity requirements: Probability of a safety-related system satisfactorily performing the required safety functions (i.e., without failure)
- Safety Integrity Level and component fault rates
  - SIL 4: 10^{-8} ...10^{-9} faults per hour
  - Typical electronic components: 10^{-5}...10^{-6} faults/hour ???
  - Typical software: 1..10 faults per 1000 line of code
- AI?

Goals

- Requirements in critical systems:
  Safety, dependability
- Architecture design (patterns) in critical systems
- Focus: Design of system architecture to ...
  - Maintain safety
  - Even in the case if AI fails
- Fault-tolerant computing has 40+ years of experience building dependable systems out of unreliable components



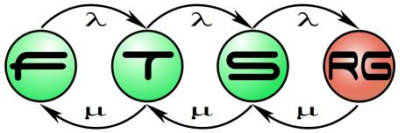
Objectives of architecture design



- Stopping (switch-off) is a safe state
- In case of a detected error the system has to be stopped
- Error detection is required

- Stopping (switch-off) is not a safe state
- Service is needed even in case of a detected error
  - full service
  - degraded (but safe) service
- Fault tolerance is required

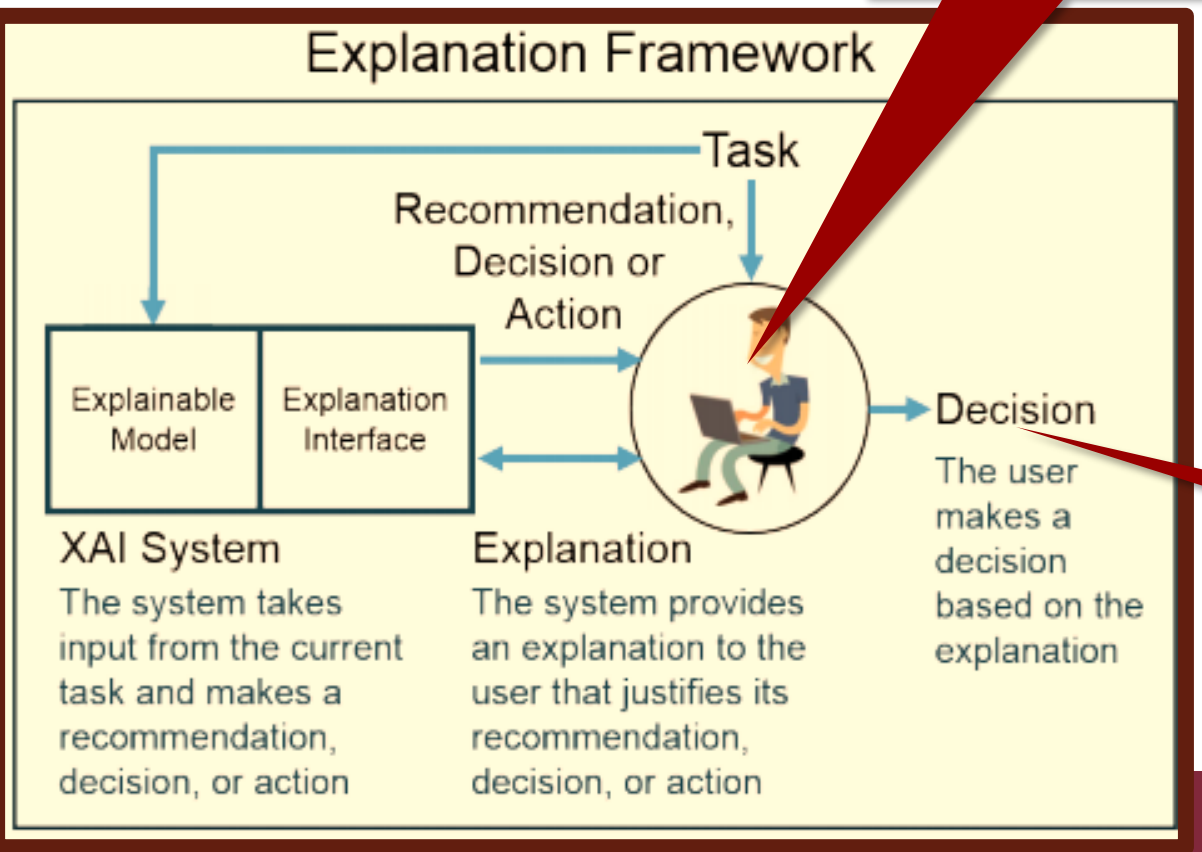
Typical architectures for fail-stop operation

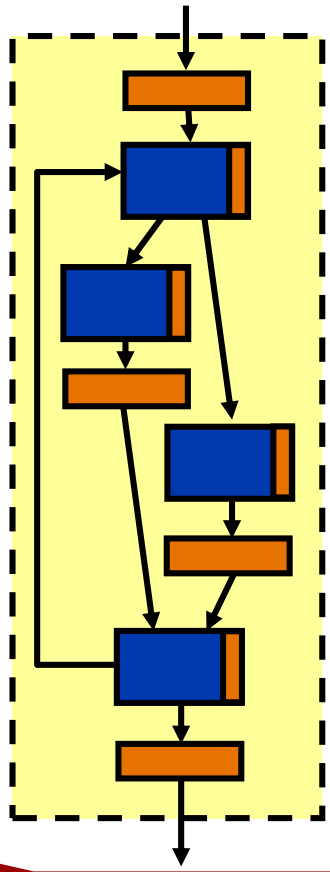


Single channel architecture with built-in self-test

- Single processing flow with error detection
- Online self-checking
  - Typically application dependent
  - Data acceptance rules,

Easier to check

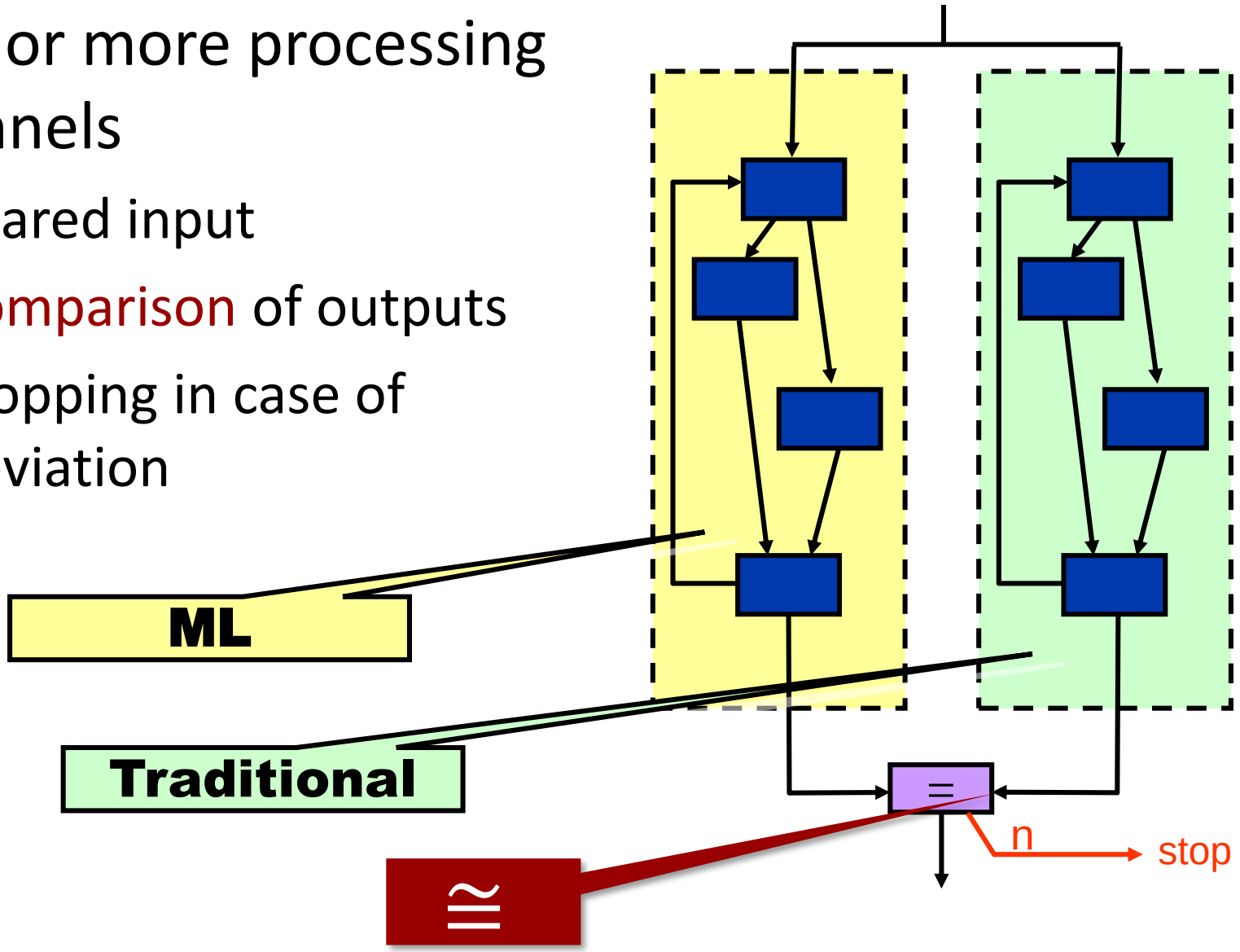




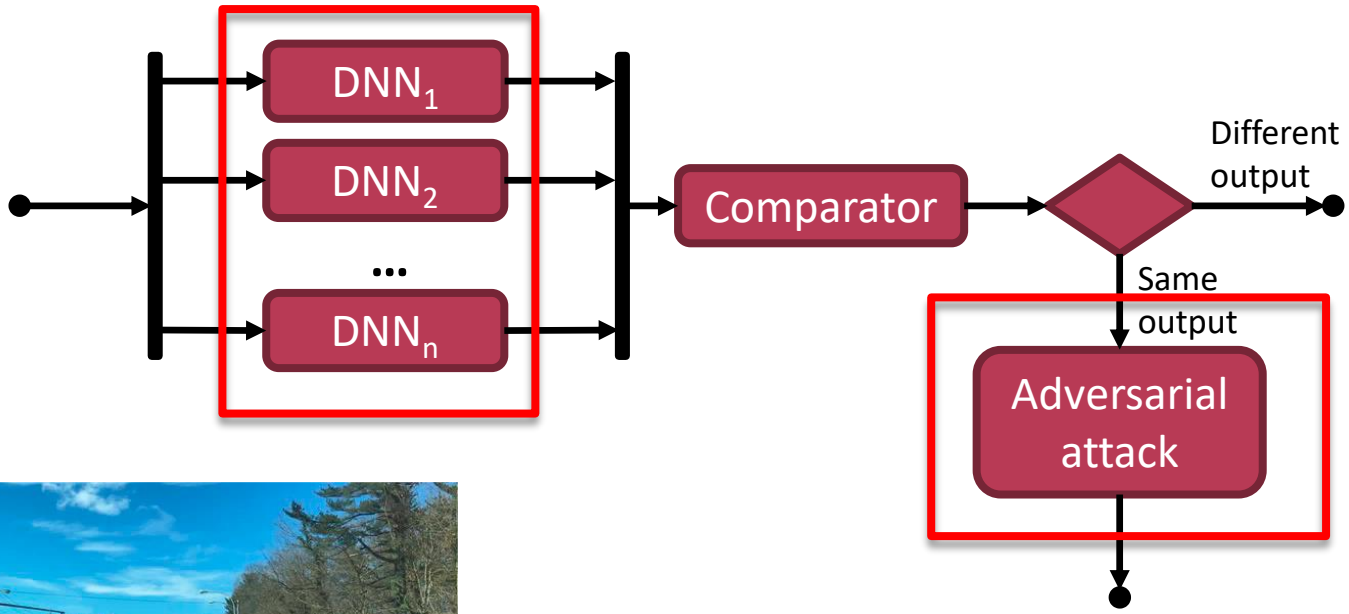
User → checking automaton

Two-channels architecture with comparison

- Two or more processing channels
  - Shared input
  - Comparison of outputs
  - Stopping in case of deviation

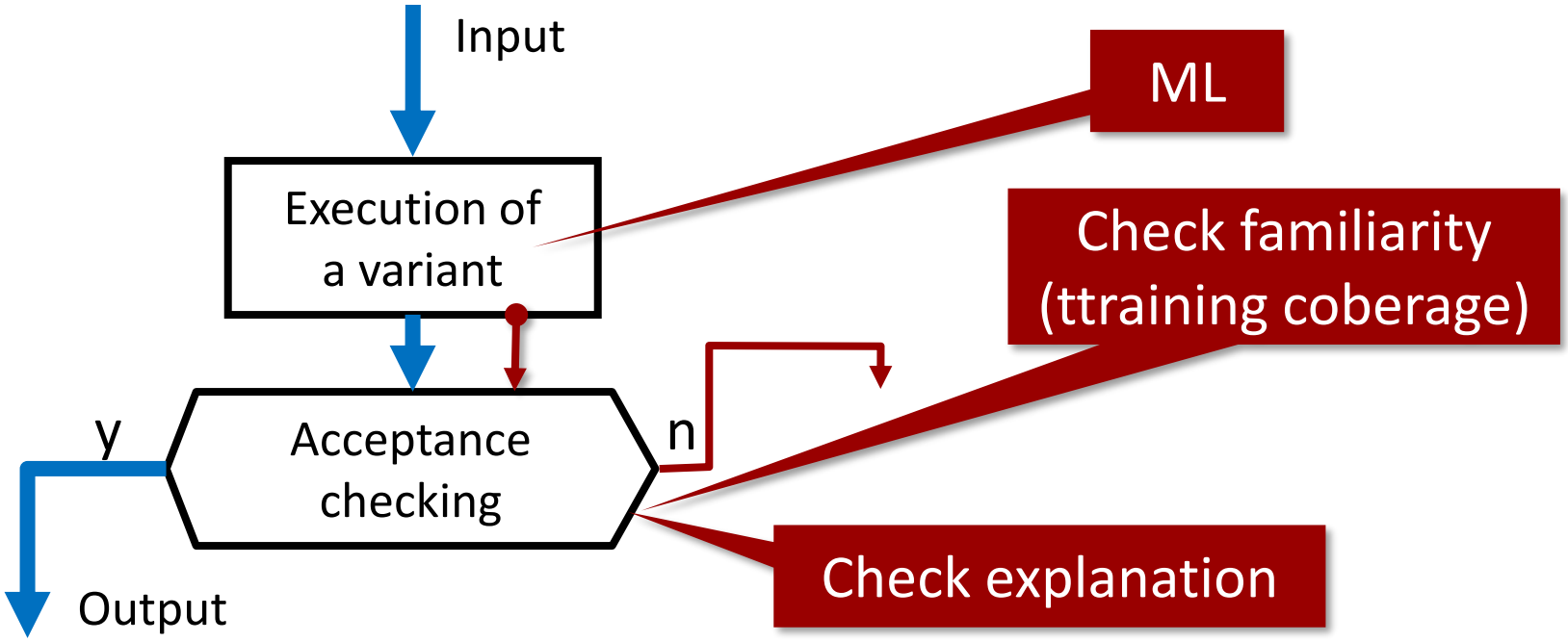


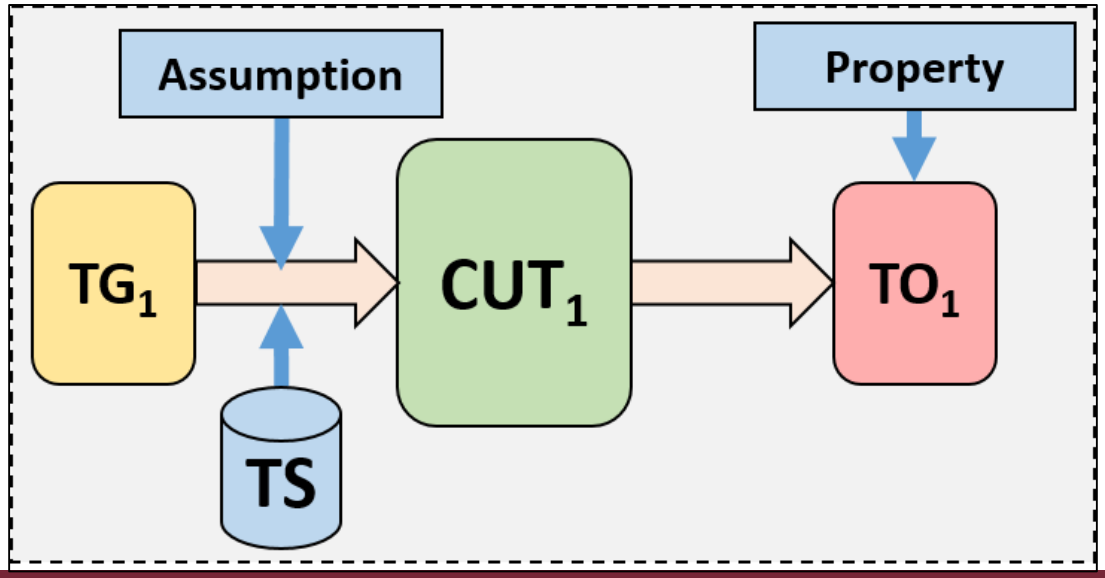
N-version


DNN1
DNN2
...
DNNn
Comparator
Different output
Same output
Adversarial attack

Recovery blocks

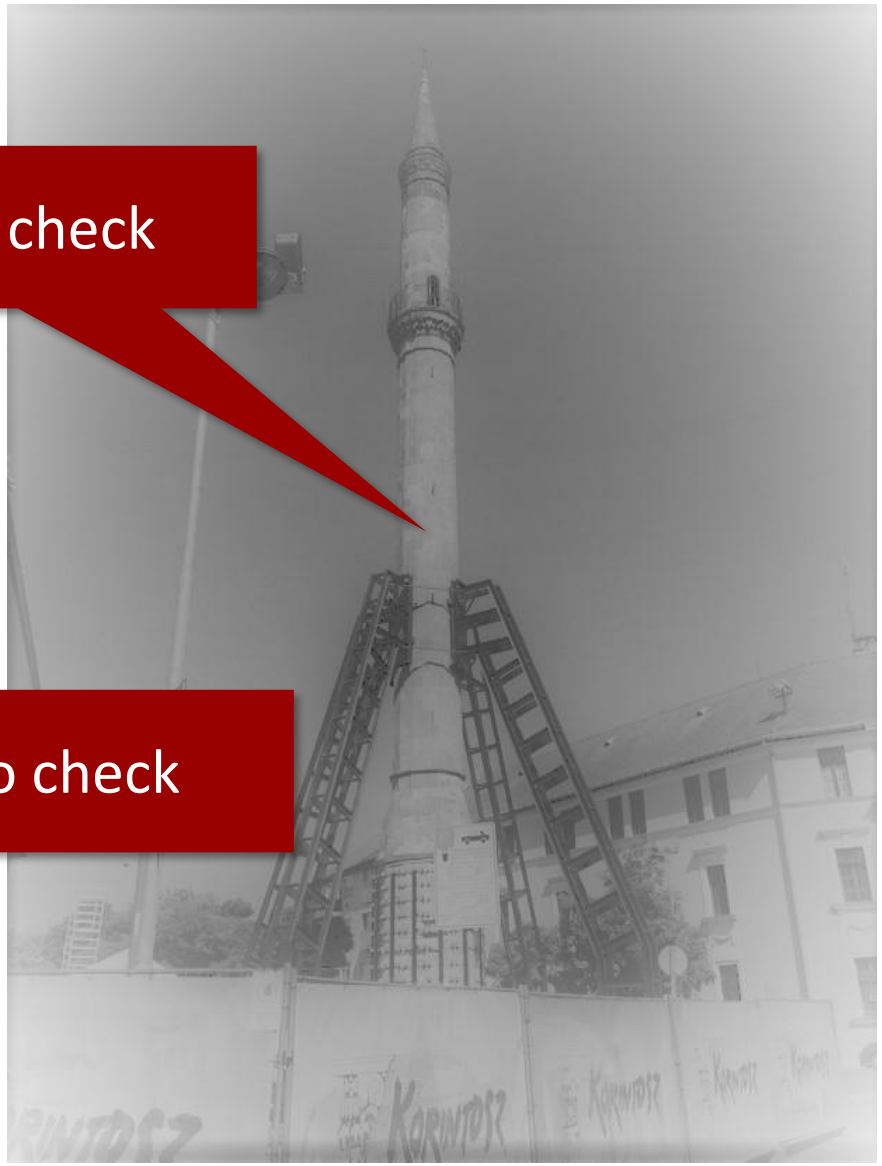
- Passive redundancy: Activation only in case of faults
  - The primary variant is executed first
  - Acceptance checking on the output of the variants
    EXPLANATION
  - In case of a detected error another variant (TRADIONAL) is executed





Wrappers as by-products of testing?

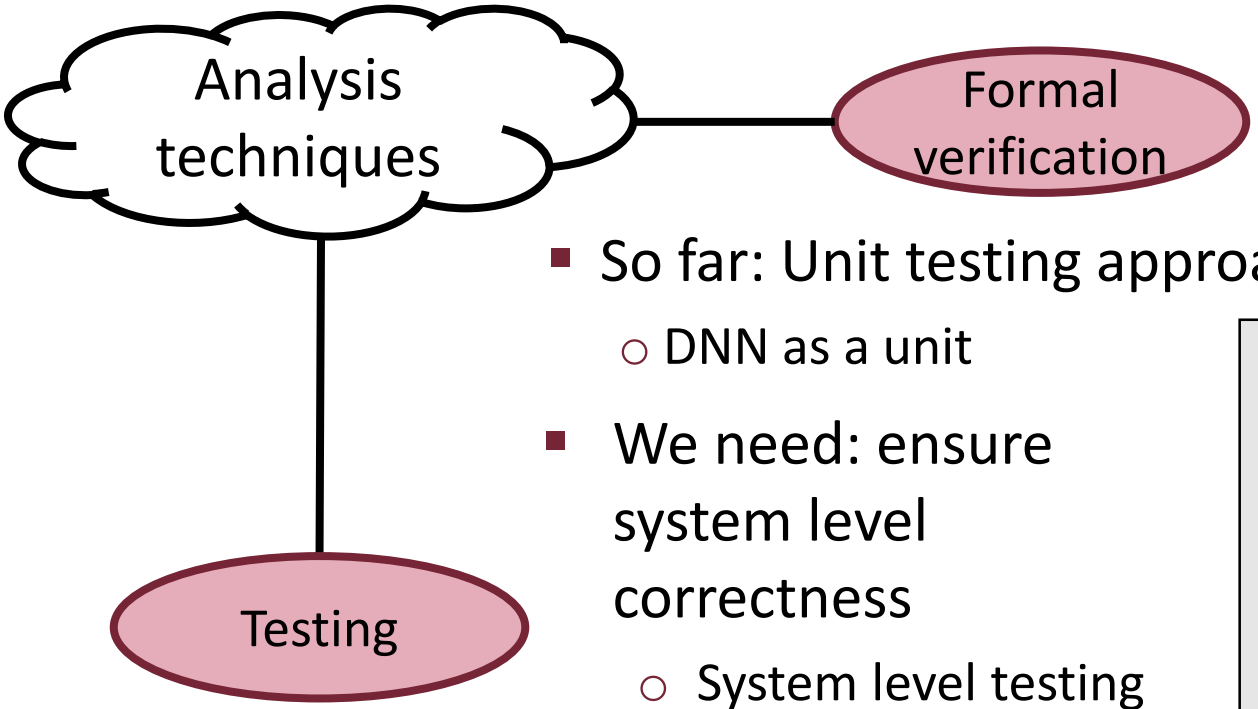
Example assumption: data quality



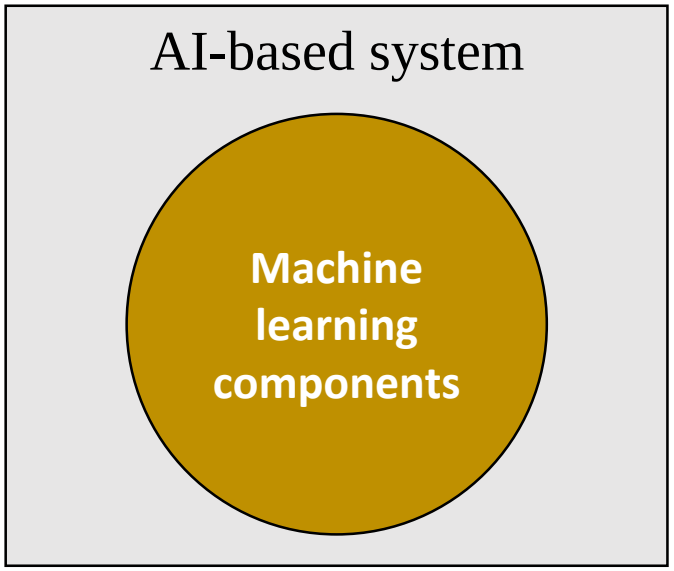
Hard to check

Easy to check

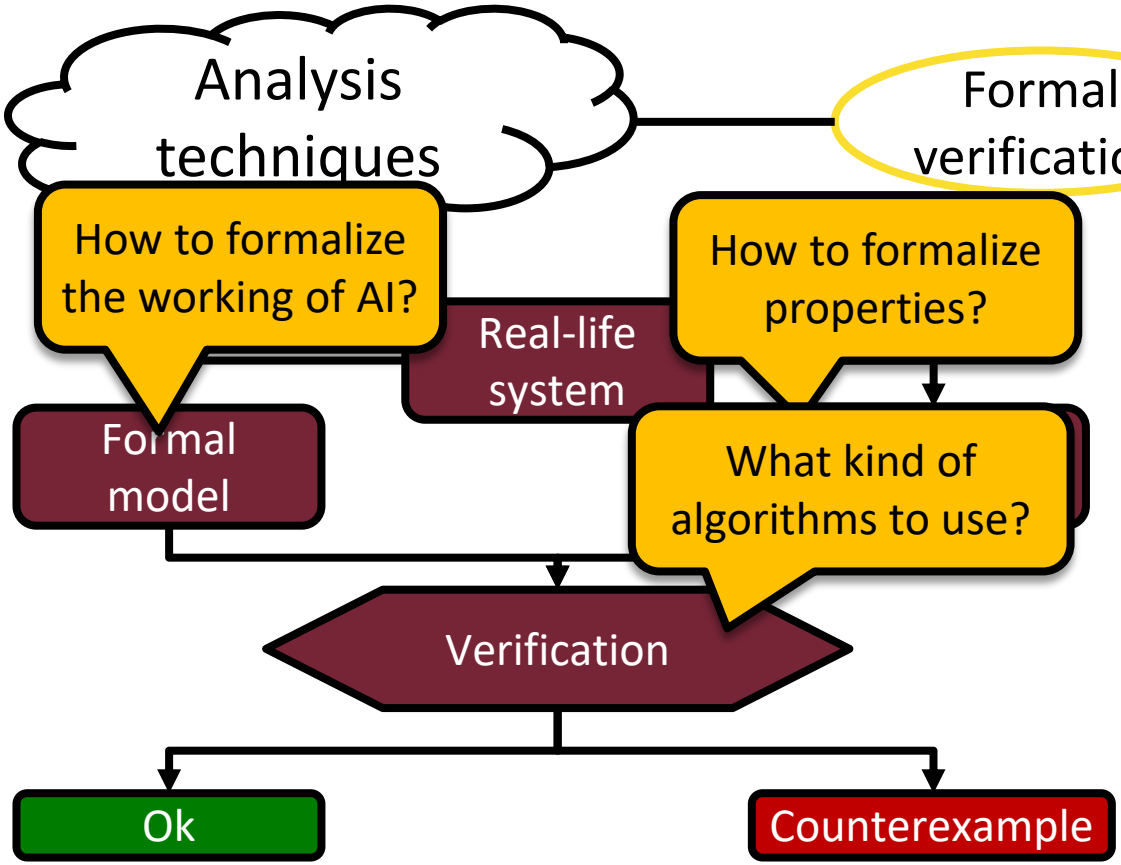
Analysis techniques overview

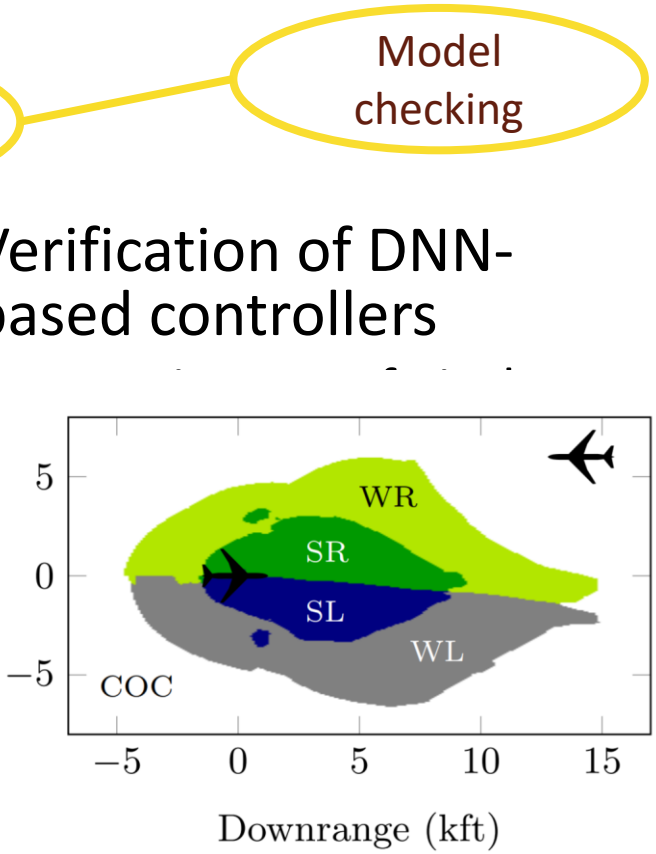


- So far: Unit testing approaches
  - DNN as a unit
- We need: ensure system level correctness
  - System level testing
  - Robustness testing
  - Extreme cases



Analysis techniques overview

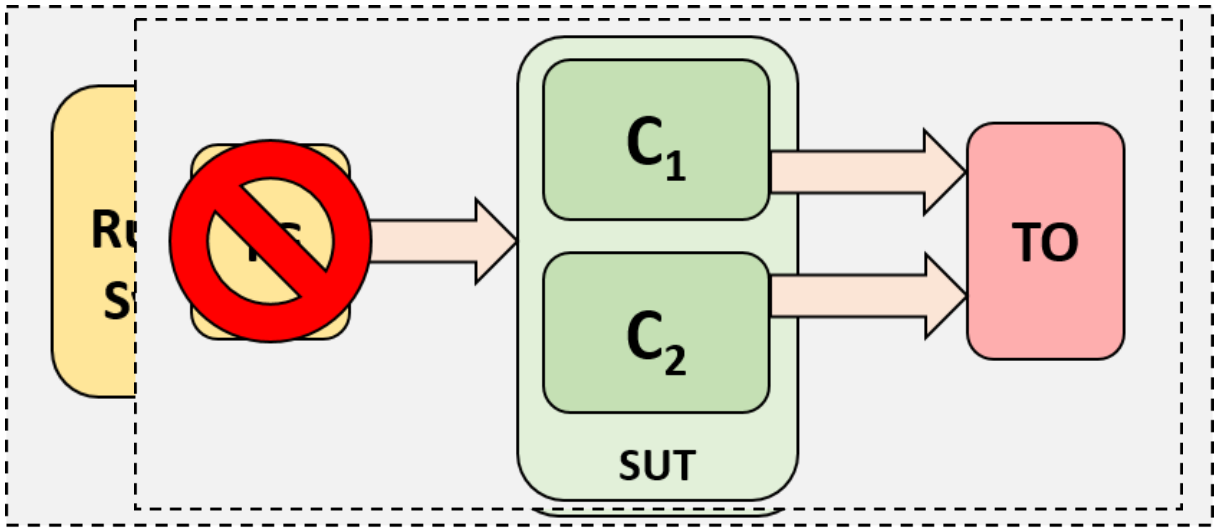




- Verification of DNN-based controllers

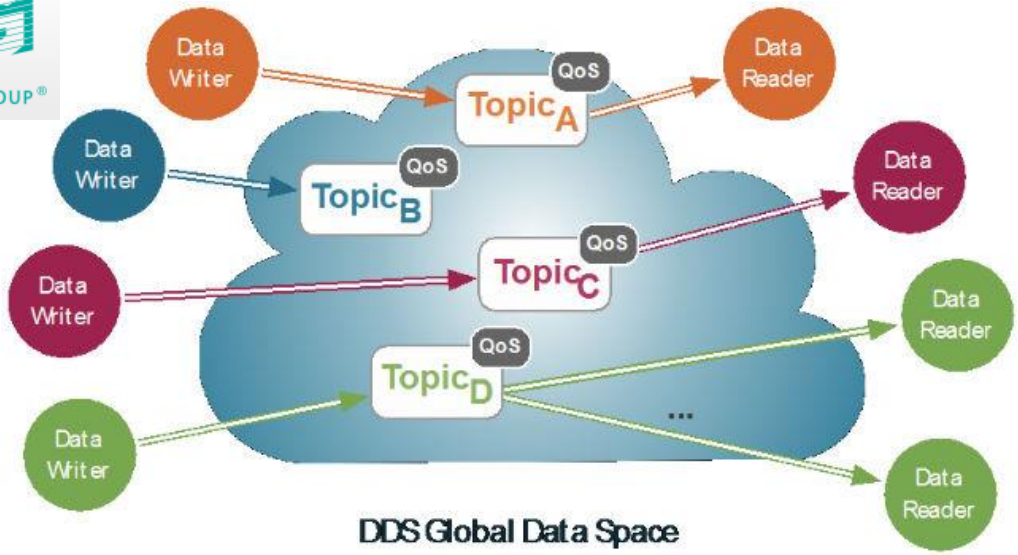
Run-time verification

- Test Generator
  - Running System
- System Under Test
  - System Components
- Test Oracle



- OMG Standard
- Publish-subscribe
- 15 QoS properties

OBJECT MANAGEMENT GROUP



- Extensions

rti Your systems. Working as one.

Summary

Summary

- Proper wrappers can eliminate ML induced failures
  - Fallback to traditional
- Checking explanations is easier
- Fault -tolerant computing has libraries