SE Considerations for Intelligent Systems

Dr. Lotfi ben Othmane
University of North Texas

Administrative

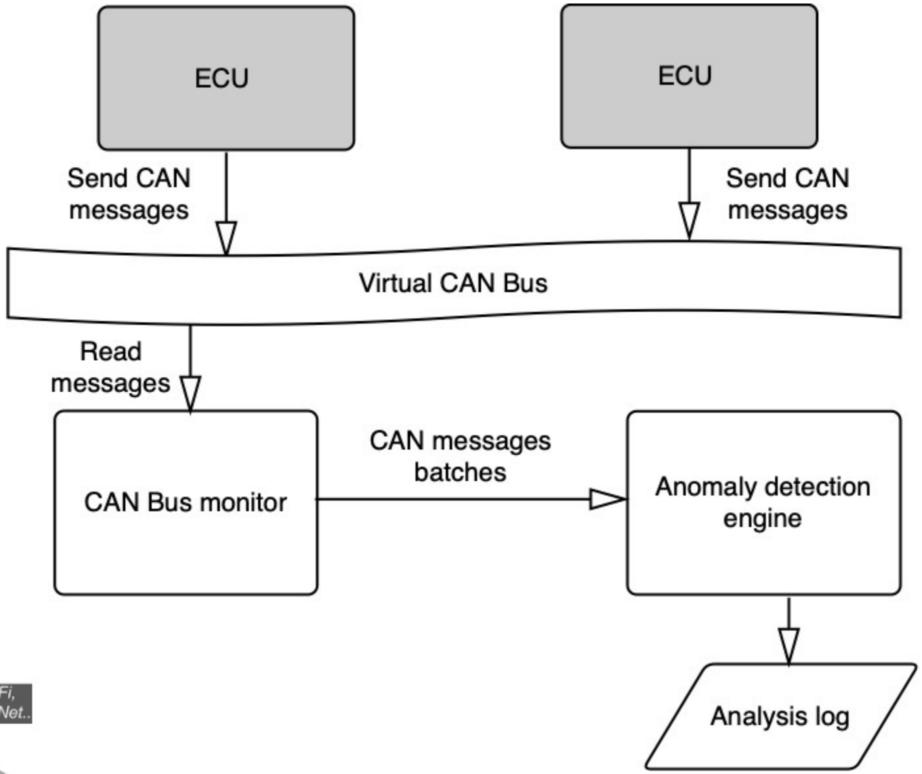
- Quiz 4 is on Tu April 2.

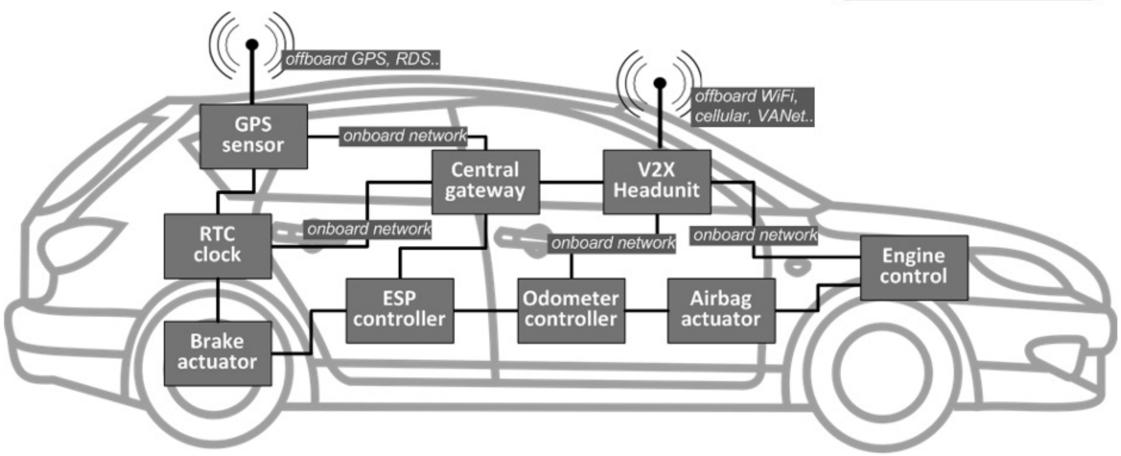
- Grading Quiz 3
- Grading Assignment 2

What is an Intelligent System?

Why this Intrusion Detection System is intelligent?

Can IDS be non-intelligent?





What is an Intelligent System?

Intelligence systems can perceive and respond to the world they operate in. Instead of relying on fixed rules, intelligent systems can gather, interpret, and reason about data.

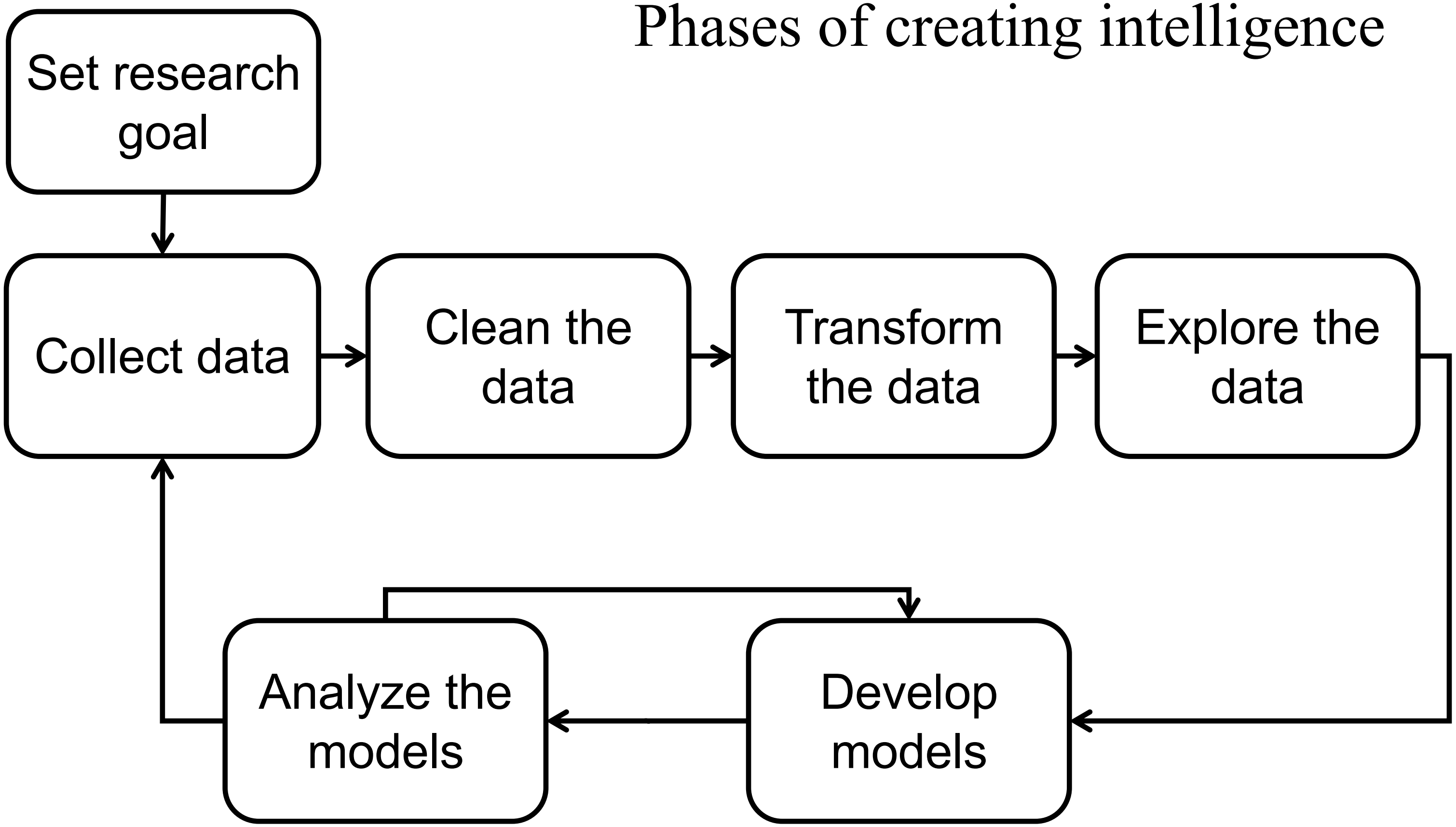
Basic Requirement for Intelligent Systems

An intelligence system

1. collects data
2. ingests new data to create/improve intelligence
3. executes the intelligence to return outcomes
4. interacts with external entities
5. orchestrates the intelligence components
6. monitors the intelligence components
7. gets telemetry about the system's performance
8. controls the behavior of the component
9. identifies runtime issues

Creating Intelligence

Phases of creating intelligence



Intelligence Runtime

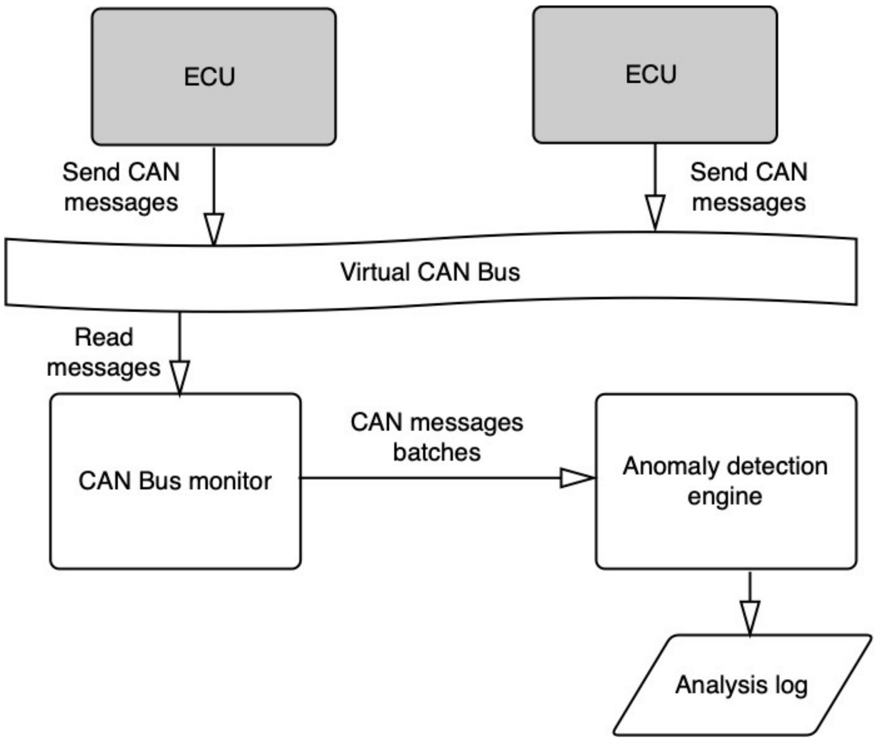
Components

1. Context
2. Features
3. Model
4. Execution engine
5. output

Sequence

1. cData = getContextData()
2. model = getIAModel()
3. oData = Predict(cData, model)
4. UpdateModel(model, cData, oData)

How does the IDS comply with this?

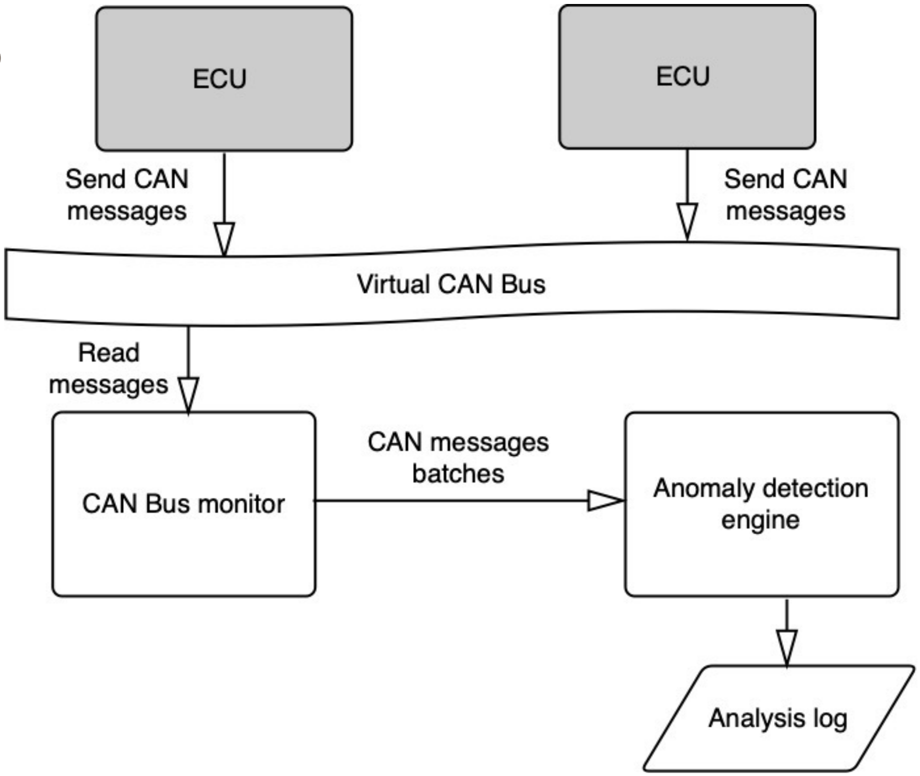


Intelligence Runtime

Often research students get enthusiastic and propose to develop IDS in the cloud

Where to put the intelligence?

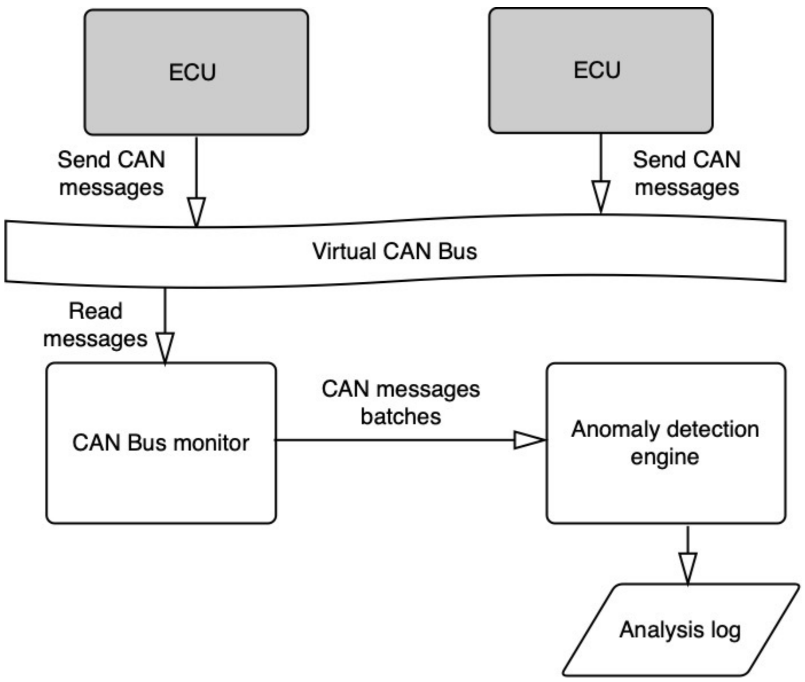
1. As a cloud service
2. In the car itself



Intelligence Runtime

Factors used to decide on the location of the intelligence

1. Latency in executing the AI
2. Latency in updating the AI
3. Impact of failure/error
4. Cost
5. Required capabilities to execute the AI
6. Impact of network failure
7. Security/threats
8. ...



Exercise: Intelligence Runtime

You are implementing a machine learning-based SPAM detection software. Should you integrate the software into:

1. the email server or
2. the email client.

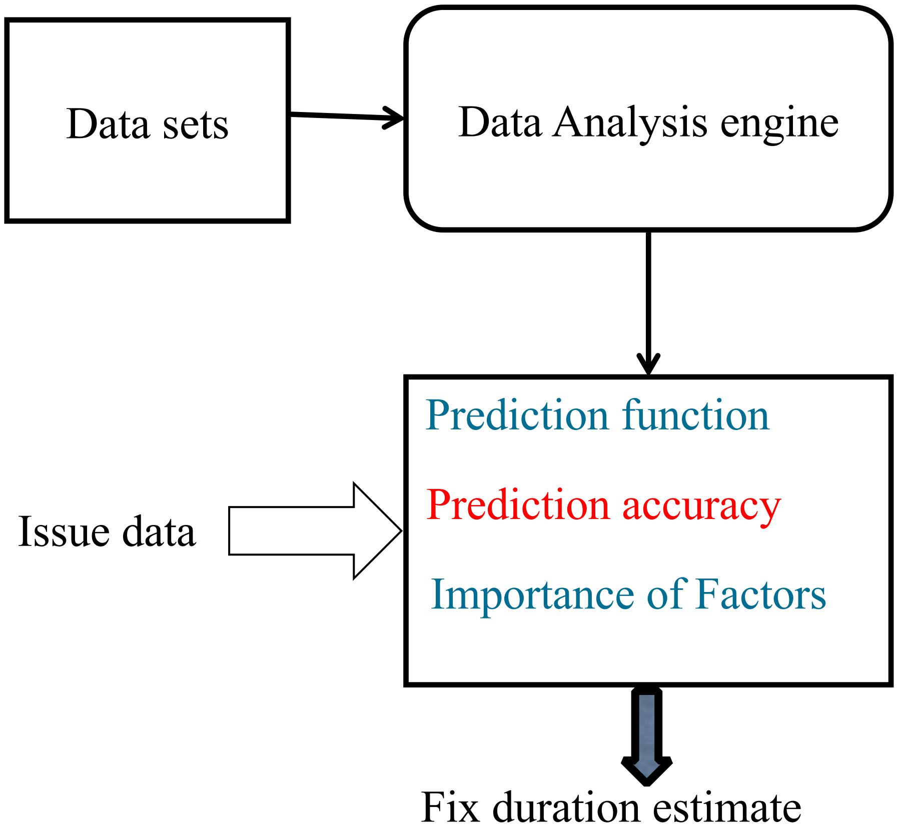
What are your decision factors?

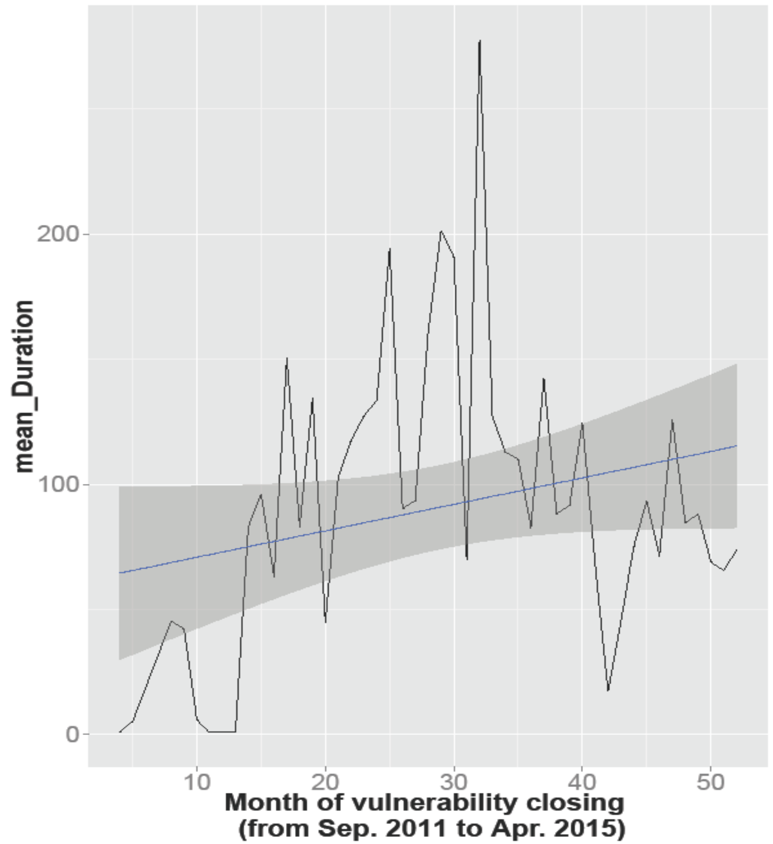
Intelligence Management

- Check SE bugs and their impact
- Check the model performance
- Check the compatibility of the AI data and context data
  - Have the dataset structure changed?
- Check the runtime constraints
  - Do we need more memory or disk space?

Intelligence Management

What can go wrong in an AI system such as this AI-based cost estimation system?





Orchestrating the AI system

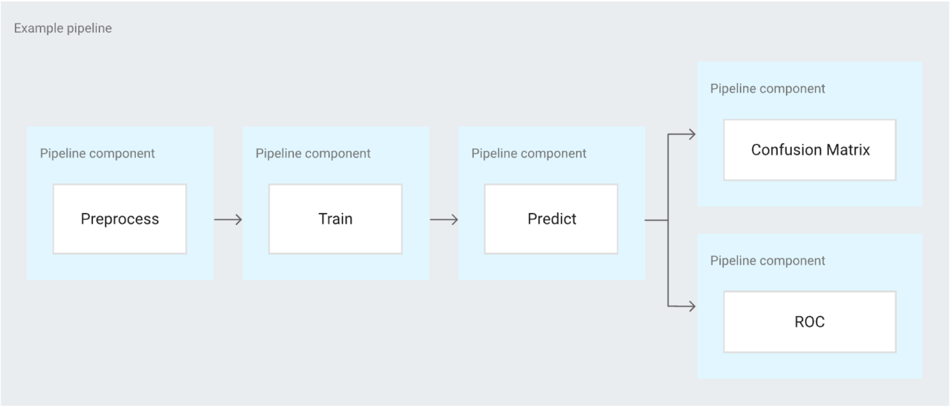
Orchestration decouples flow and sequence from processing.

Benefits:

- Control the execution of the AI components
- Provide telemetry about the execution
- Localize errors
- Scale effectively
- Degrade slowly

Recall: Properties of good architecture

- Easy to modify and grow
- Loosely coupled
- Comprehensible
- Controllable



Exercise: Assist Blind with AR

What machine learning techniques would you use?

- Qualcomm processor
- Android 11 OS.
- Stores files and programs in a 32 GB microSD card.

- Recognize objects close to the user when walking a street and inform them about their types, which could be humans, cars, trees, doors, stairs, etc.
- Direct the user when crossing a street, including telling them the direction to take to reach their specified destination (they enter the destination to the system using voice commands) and the traffic light phases,
- Inform the user about their acquaintances when they walk nearby.

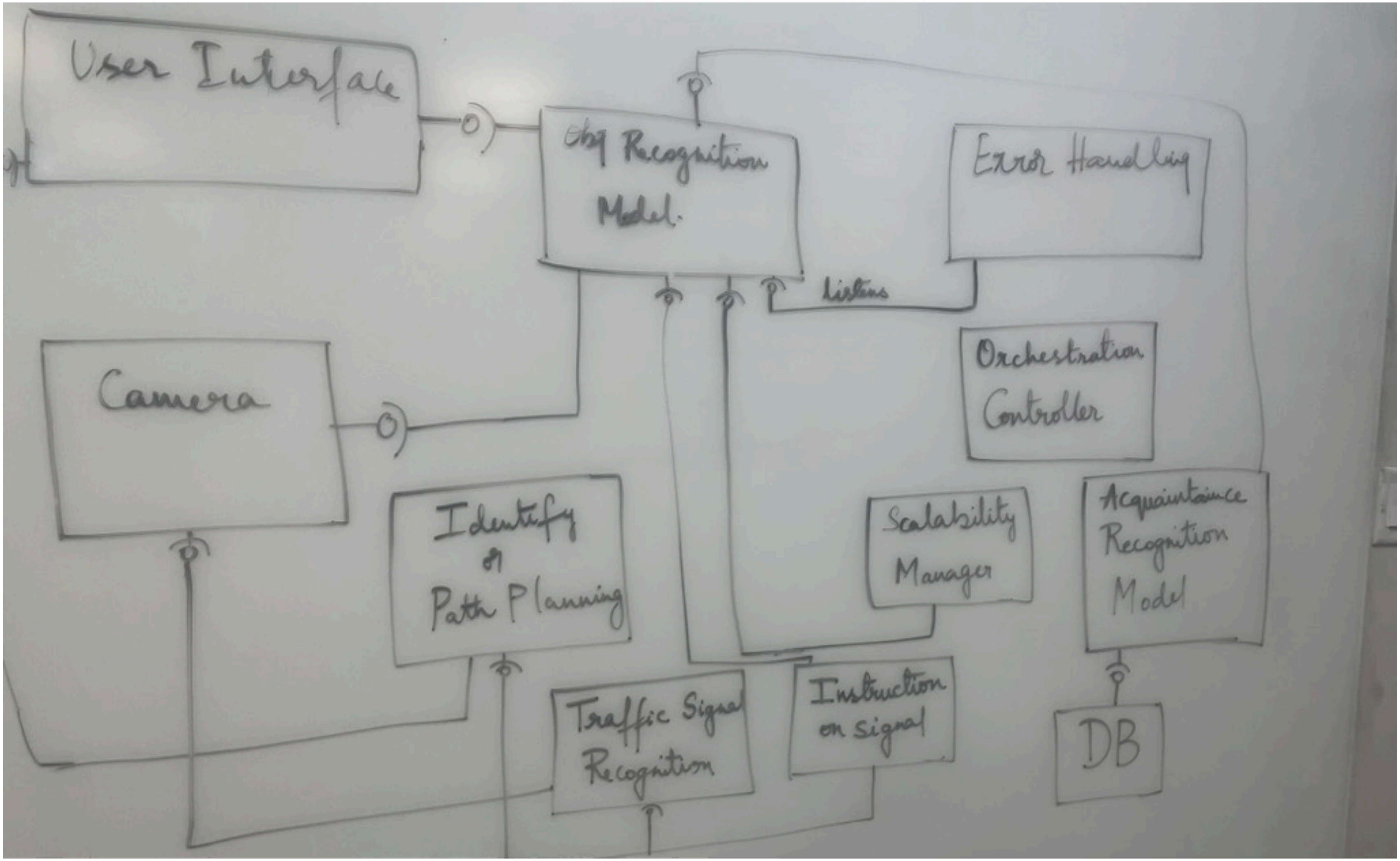
Exercise: Assist Blind with AR

Design a component diagram sketch for a system, considering:

- Orchestration
- Localize errors
- Control
- Monitor execution performance
- Scale the system

- Recognize objects close to the user when walking a street and inform them about their types, which could be humans, cars, trees, doors, stairs, etc.
- Direct the user when crossing a street, including telling them the direction to take to reach their specified destination (they enter the destination to the system using voice commands) and the traffic light phases,
- Inform the user about their acquaintances when they walk nearby.

Group work


User Interface
Obj Recognition Model.
Error Handling
listens
Camera
Orchestration Controller
Identify or Path Planning
Scalability Manager
Acquaintaince Recognition Model
Traffic Signal Recognition
Instruction on signal
DB

Architecture Drivers

The architecture drivers define the what and why about the architecture

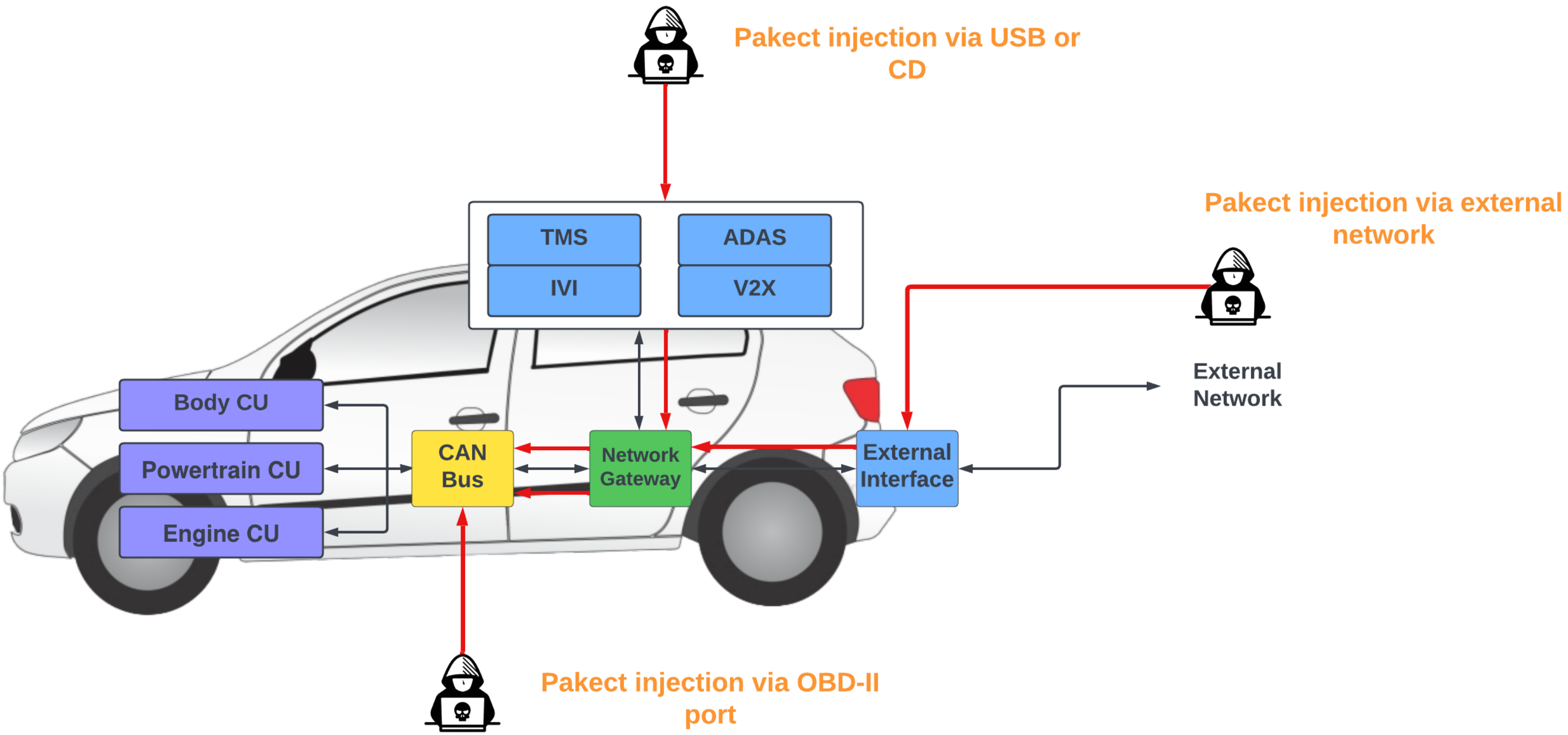
- They include:
  1. Primary functionality
  2. Design purpose
  3. Quality attributes
  4. Architectural concerns
  5. Architectural constraints

Quality Attributes (QAs)

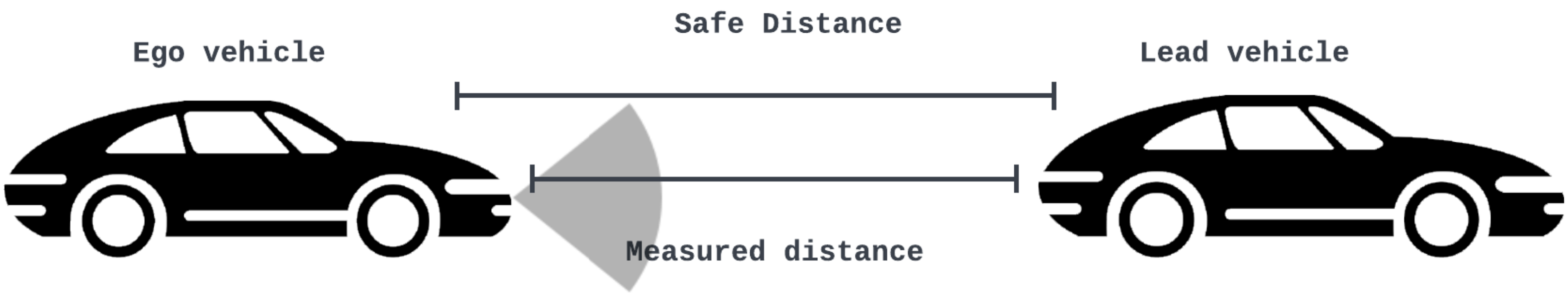
- QAs indicates how well the system satisfies the needs of the stakeholders
  - Are measurable and testable properties of the system
  - Constraints on the functional requirements

- Important functional requirements should be associated with quality attributes, e.g.,
  1. How fast should the function be?
  2. How secure should the function be?
  3. How modifiable should the function be?
  4. Etc.

Case: Intrusion Detection System

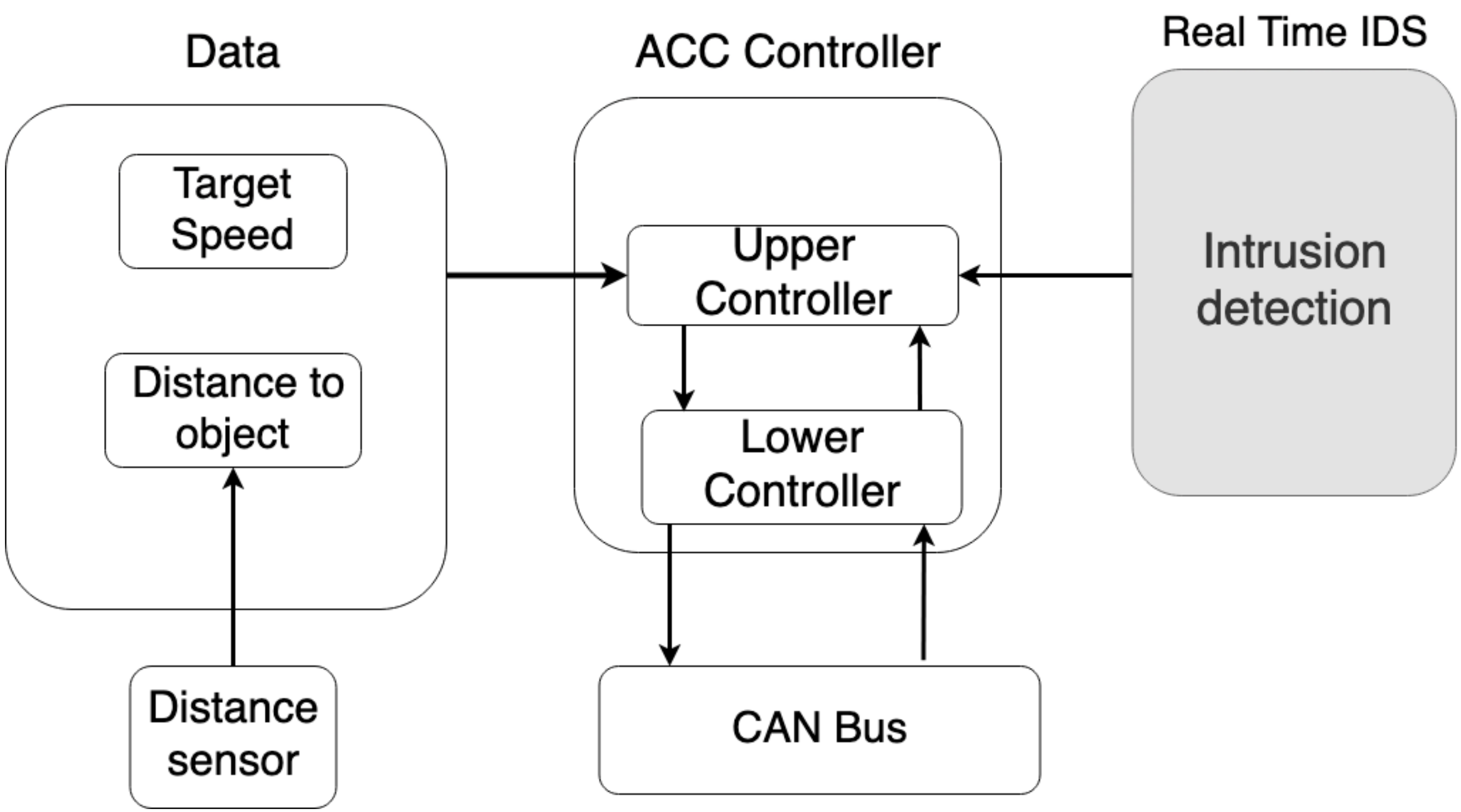


Case: Intrusion Detection System



The Adaptive Cruise Control (ACC)-equipped vehicle uses radar sensors to measure the distance to the lead vehicle, to take proper actions (acceleration or deceleration)

Case: Intrusion Detection System

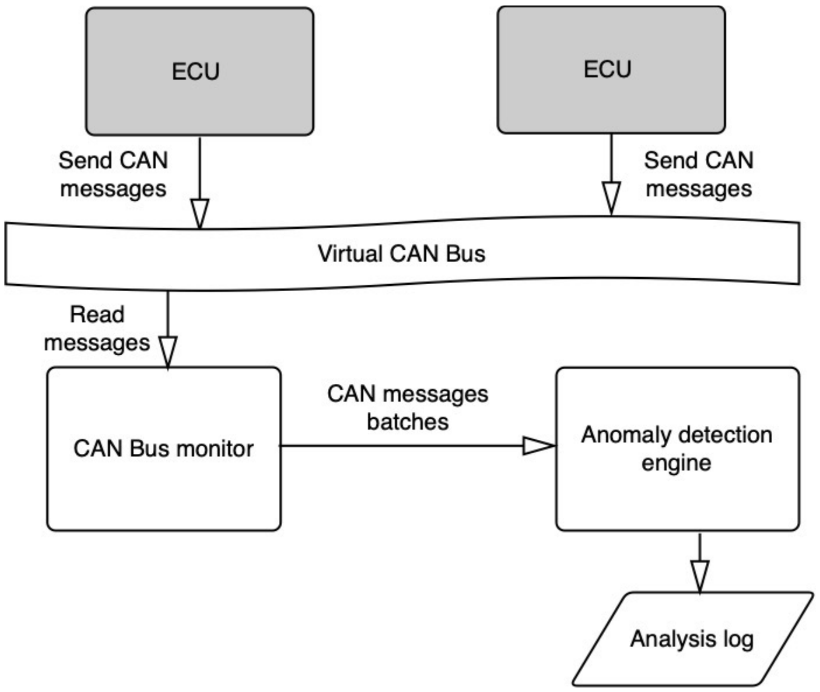


Architecture of Adaptive Cruise Control-IDS

Case: QA for ML-based IDS

Need to address the three requirements

1. Small response time
2. Must not loose CAN data
3. Run on an ECU that has limited capabilities



Case: Constraints for ML-based IDS

Constraint	Value
Recommended maximum rate of injection of CAN message	1908/sec
Rate of injection of CAN message	1000/sec
Reaction time constraint	2.5 sec
Detection speed of 1000 messages using the similarity threshold technique in offline	0.0025 sec

Exercise: Assist Blind with AR

Identify QAs requirements that apply to the system

- Qualcomm processor
- Android 11 OS.
- Stores files and programs in a 32 GB microSD card.

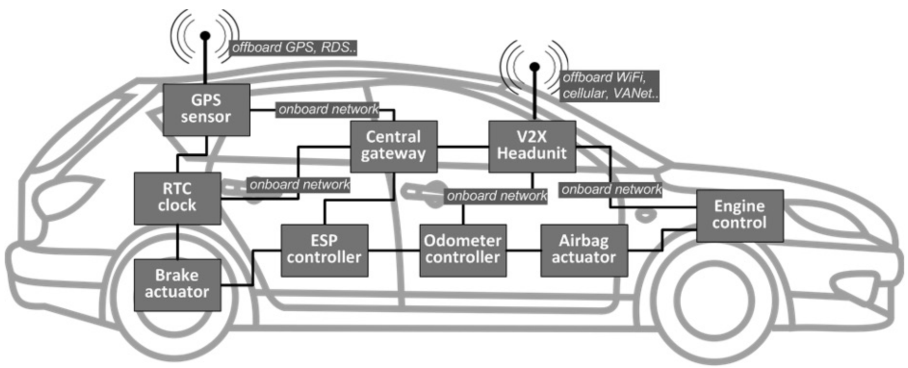
- Recognize objects close to the user when walking a street and inform them about their types, which could be humans, cars, trees, doors, stairs, etc.
- Direct the user when crossing a street, including telling them the direction to take to reach their specified destination (they enter the destination to the system using voice commands) and the traffic light phases,
- Inform the user about their acquaintances when they walk nearby.

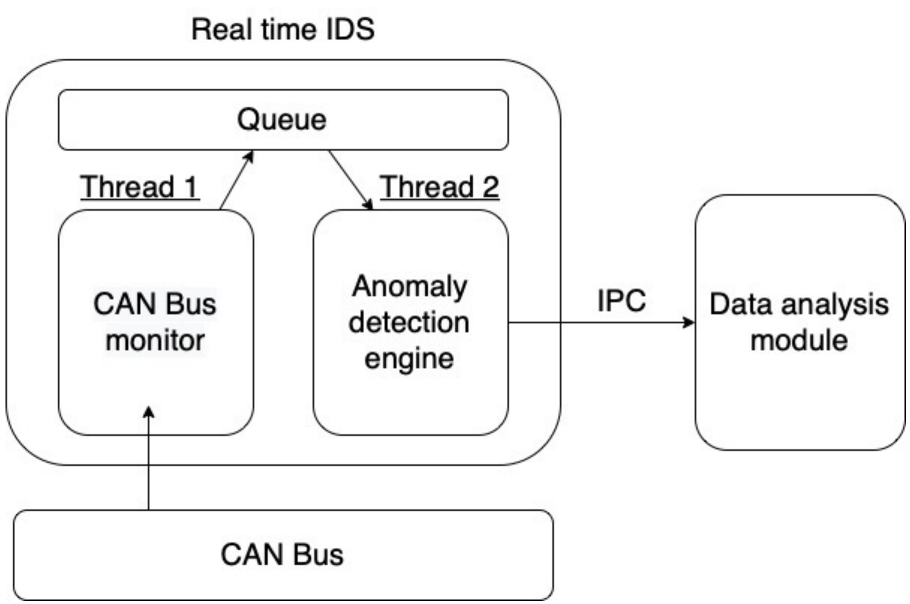
Architecture Patterns and Tactics

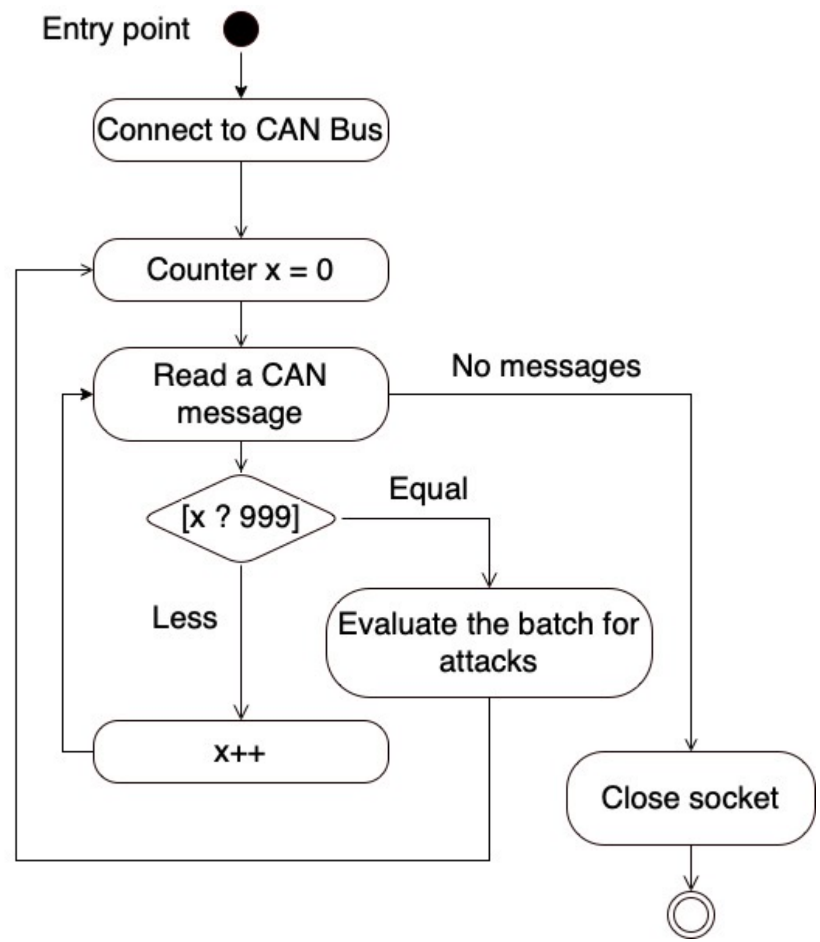
- Architectural/design patterns are conceptual solutions for recurring problems
- Tactics are design decisions that influence the control of a quality attribute response

- Patterns, tactics, and styles are tricks to solve architecture problems.

Example 1: Concurrency for-IDS



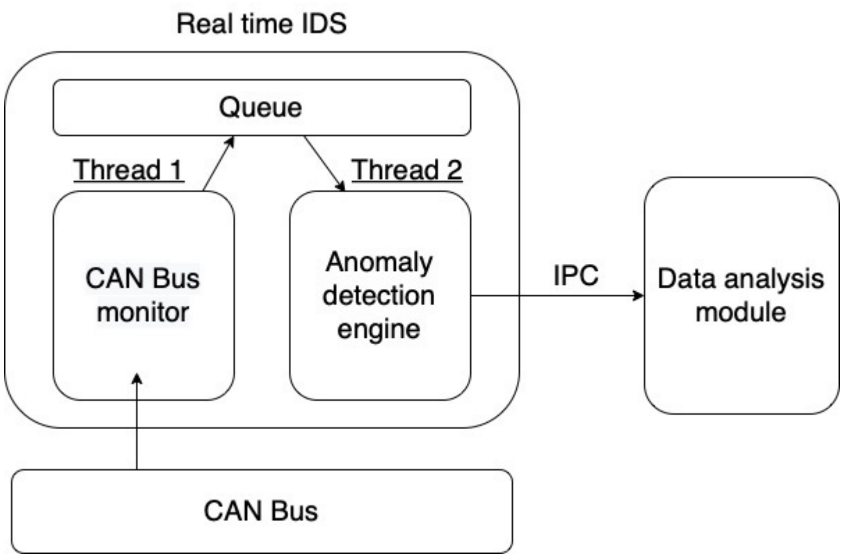




Example1: QA for ML-based IDS

Need to address the three requirements

1. Small response time
2. Must not loose CAN data
3. Run on an ECU that has limited capabilities



Example 1: Concurrency for-IDS

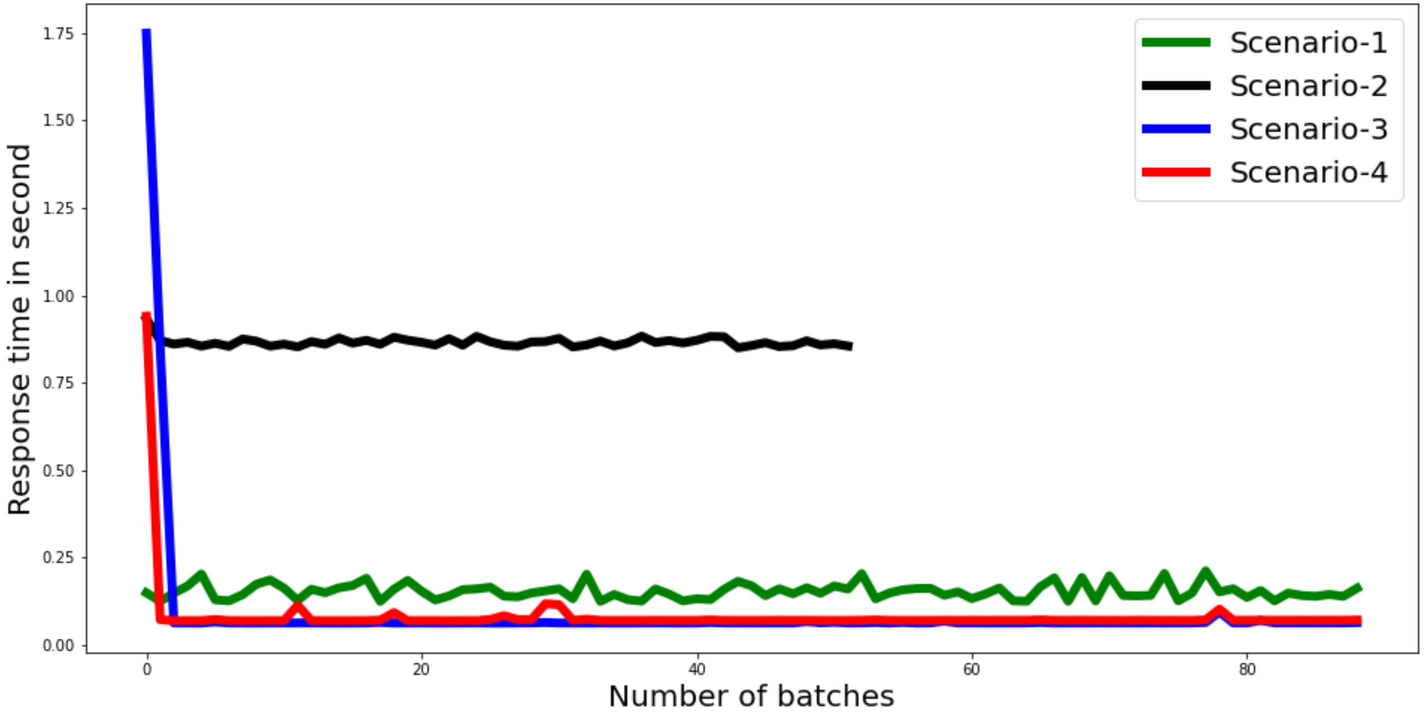
- ECU emulator sends injection of speed-reading CAN messages

Anomaly evaluation times and response times for four architecture scenarios

Architecture	Average time of sending 1000 CAN messages	Average evaluation time	Response time
Scenario 1-single process	998 ms	152 ms	1.15 sec
Scenario 2 – single with subprocess	944 ms	865 ms	1.809 sec
Scenario 3 – single process with two threads	950 ms	90 ms	1.04 sec
Scenario 4 – two processes	945 ms	81 ms	1.026 sec

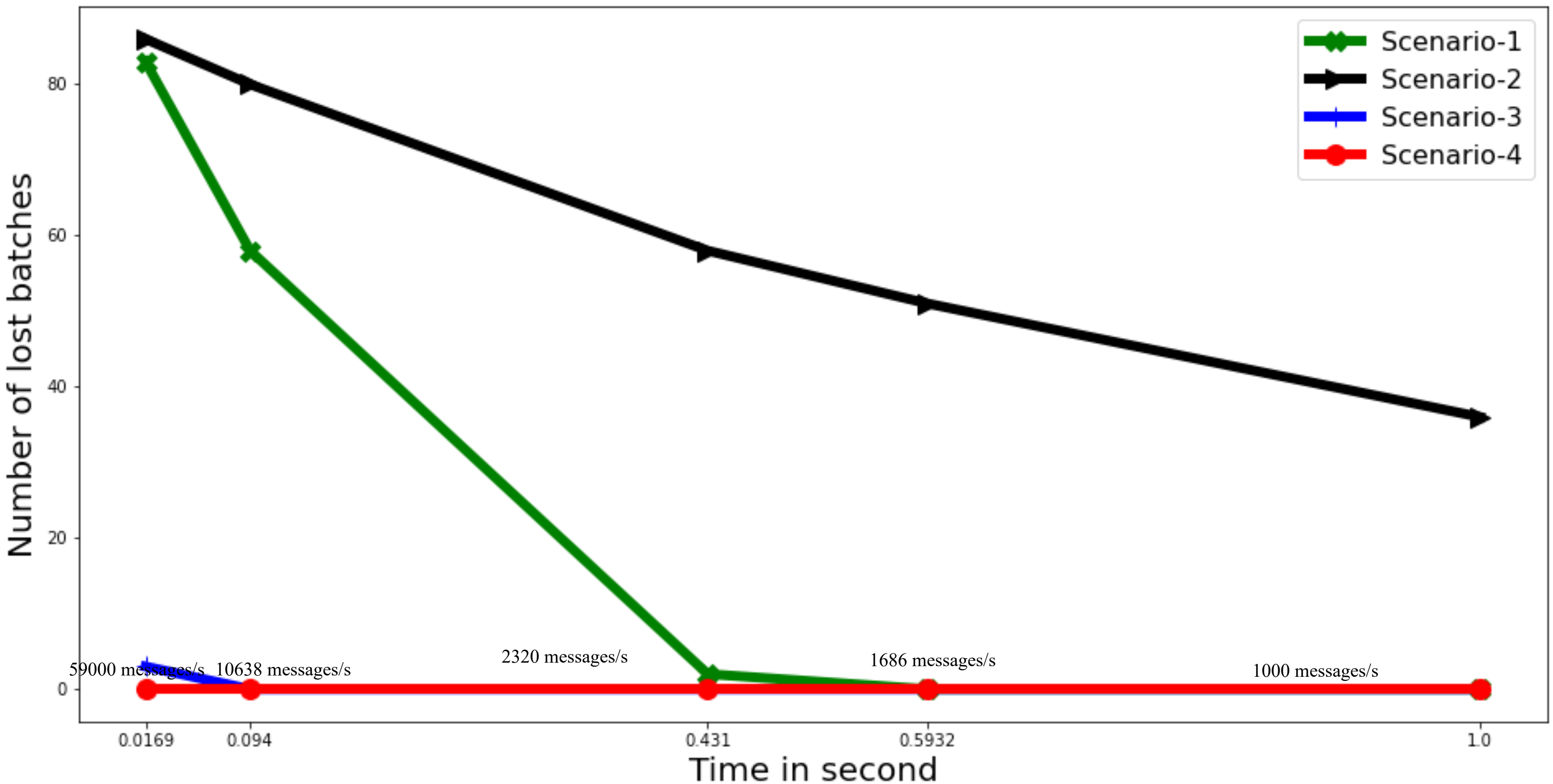
Example 1: Concurrency for-IDS

- Injected 88,000 speed reading CAN messages



Anomaly Evaluation Time of the Four Architecture Scenarios

Analysis of the Reliability of the Architecture Scenarios in Terms of CAN Message Losses



Ratio of messages losses vs speed of sending 1000 CAN bus in four architecture scenarios

Example 2- Digital Twins

Goal: Working offline with an intelligent system.

A digital twin is a virtual representation of an object or system that spans its lifecycle, is updated from real-time data, and uses simulation, machine learning and reasoning to help decision-making.

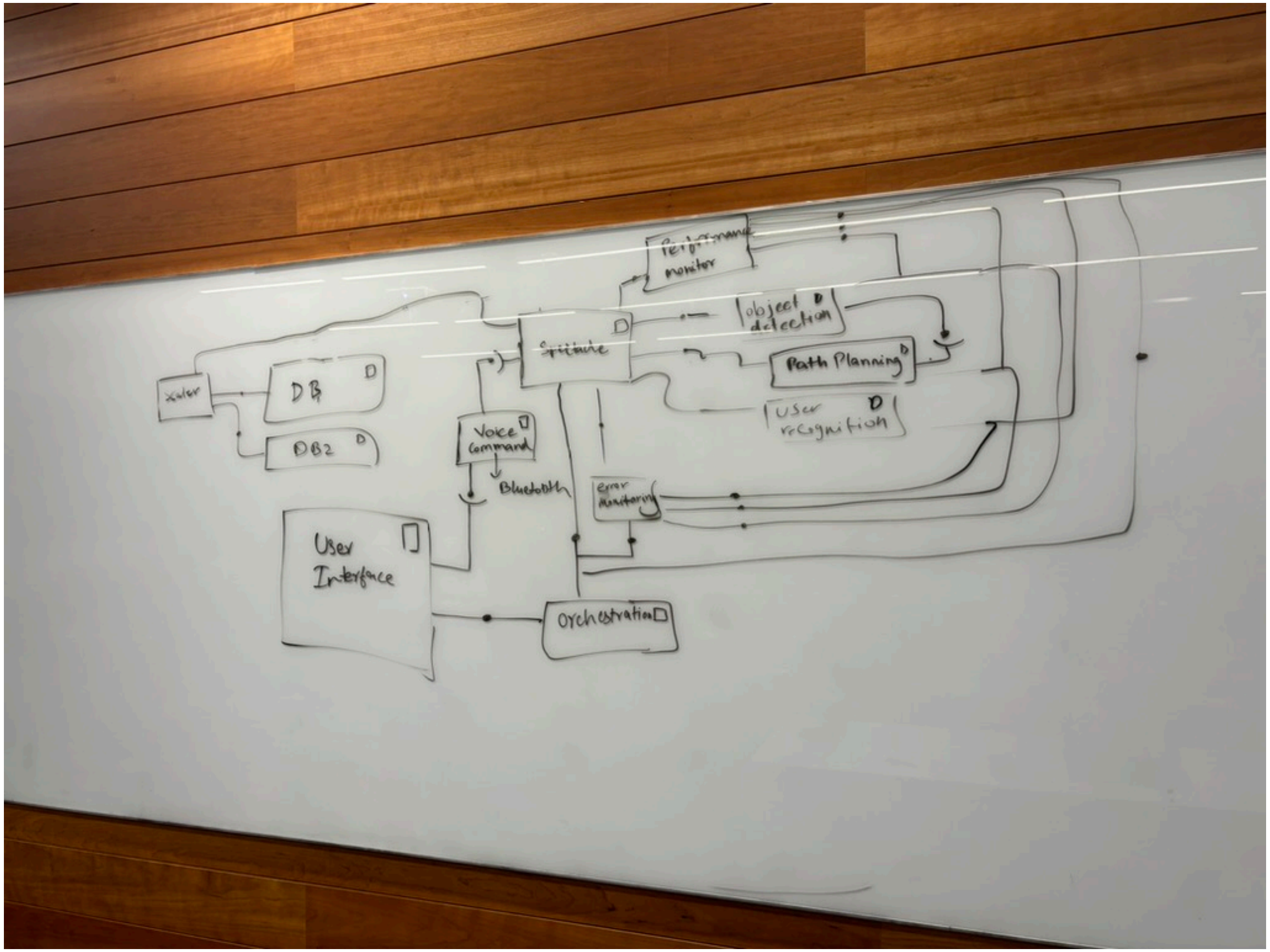
Exercise: Assist Blind with AR

- Qualcomm processor
- Android 11 OS.
- Stores files and programs in a 32 GB microSD card.

- Recognize objects close to the user when walking a street and inform them about their types, which could be humans, cars, trees, doors, stairs, etc.
- Direct the user when crossing a street, including telling them the direction to take to reach their specified destination (they enter the destination to the system using voice commands) and the traffic light phases,
- Inform the user about their acquaintances when they walk nearby.

Exercise: Assist Blind with AR

Update your component diagram sketch to account for Response time constraint and need to support different AR devices



Summary

An intelligence system

1. collects data
2. ingests new data to create/improve intelligence
3. executes the intelligence to return outcomes
4. interacts with external entities, and
5. orchestrates the intelligence components
6. monitors the intelligence components
7. gets telemetry about the system’s performance
8. controls the behavior of the component
9. identifies runtime issues

Thank you

Any Question?