

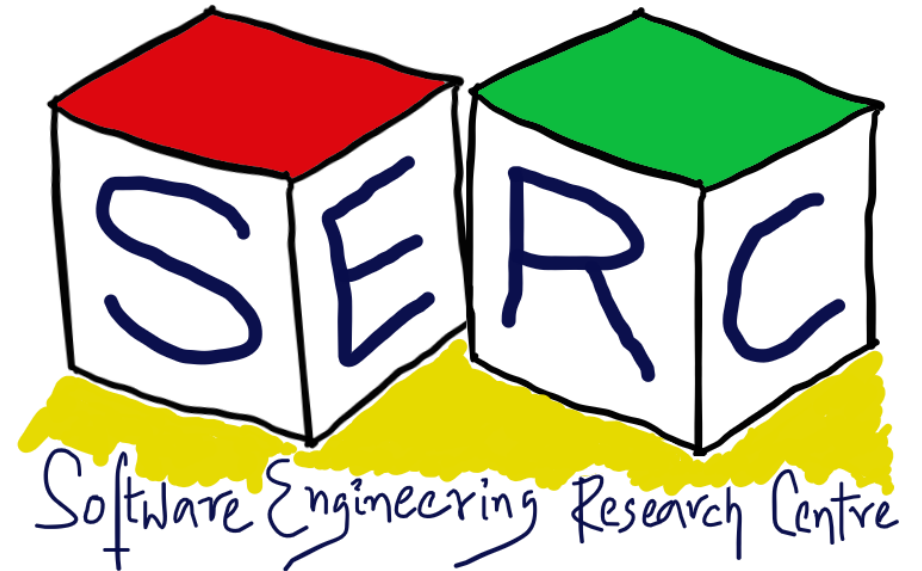
# Architectural Framework

**CS6.401 Software Engineering**

Dr. Karthik Vaidhyanthan

[karthik.vaidhyanthan@iiit.ac.in](mailto:karthik.vaidhyanthan@iiit.ac.in)

<https://karthikvaidhyanthan.com>



INTERNATIONAL INSTITUTE OF  
INFORMATION TECHNOLOGY

HYDERABAD

# Acknowledgements

The materials used in this presentation have been gathered/adapted/generate from various sources as well as based on my own experiences and knowledge -- Karthik Vaidhyanathan

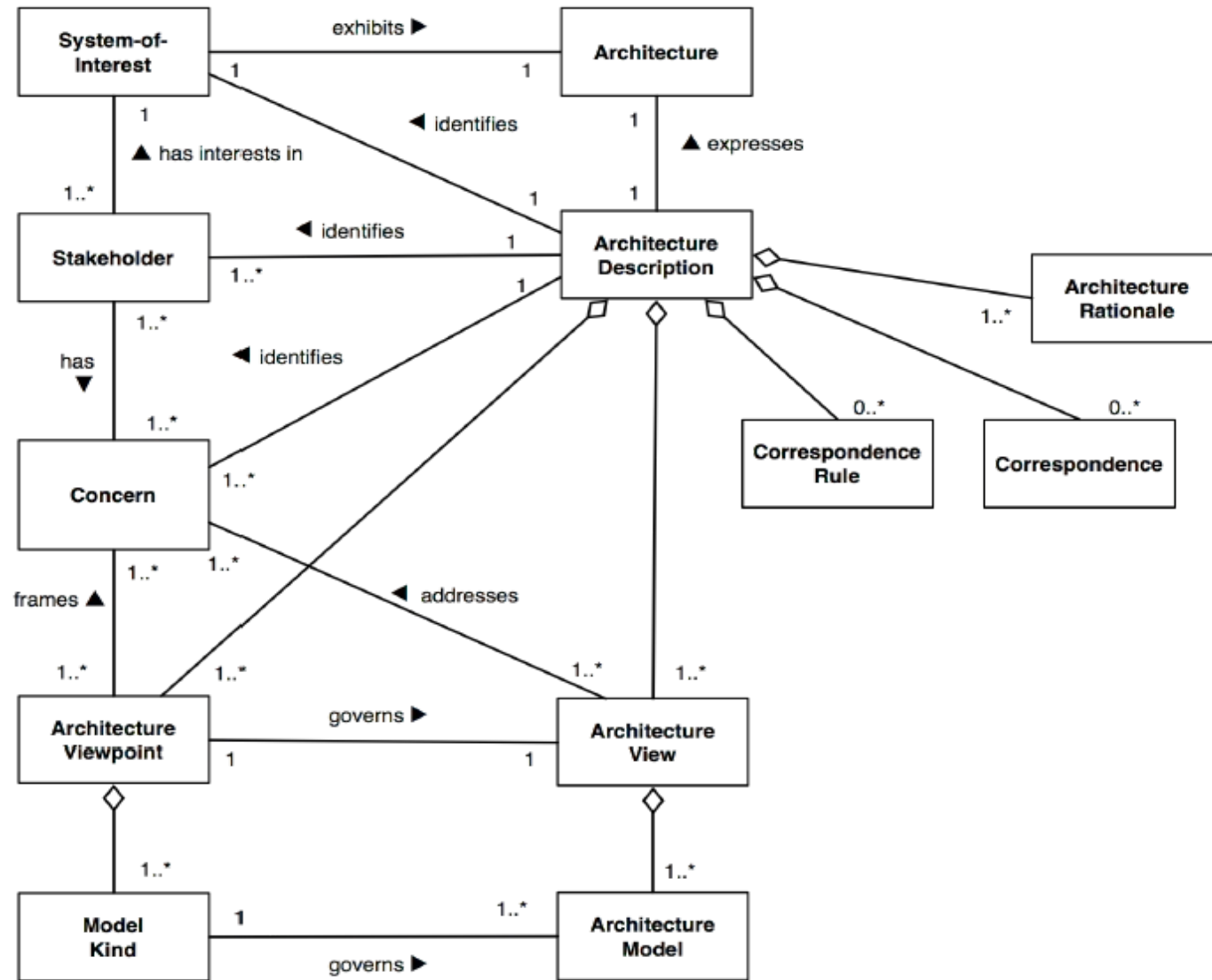
Sources:

1. Software Architecture in Practice, Len Bass, 3<sup>rd</sup> edition
2. ISO/IEC/IEEE 42010, Systems and Software Engineering – Architecture Description



# Architecture Description

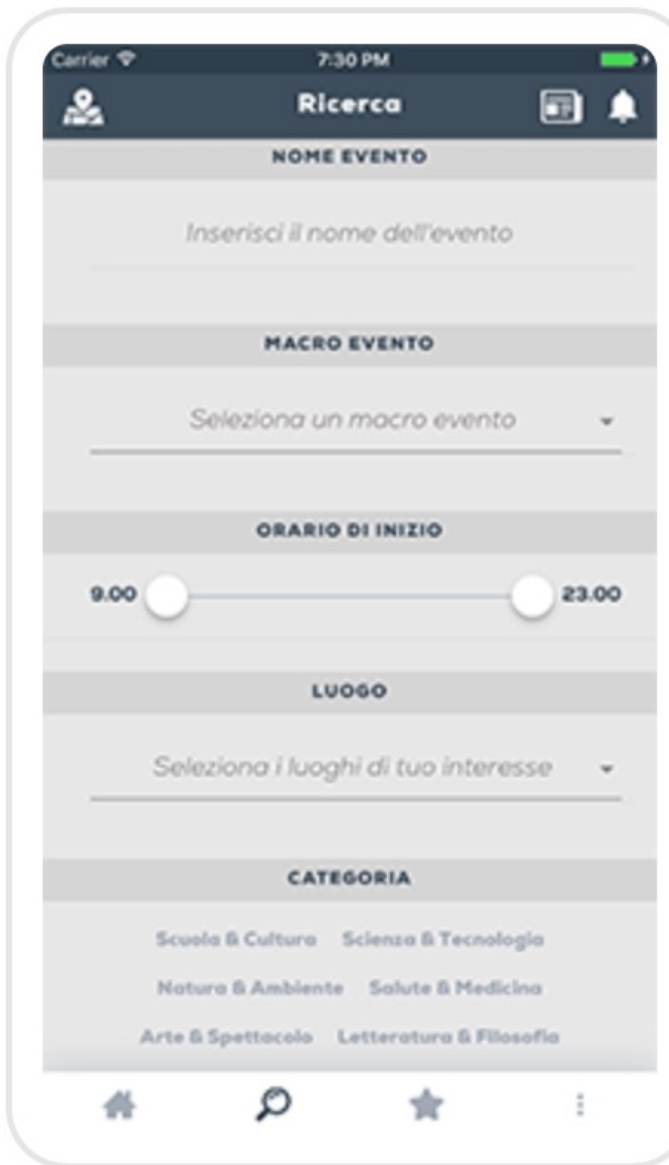
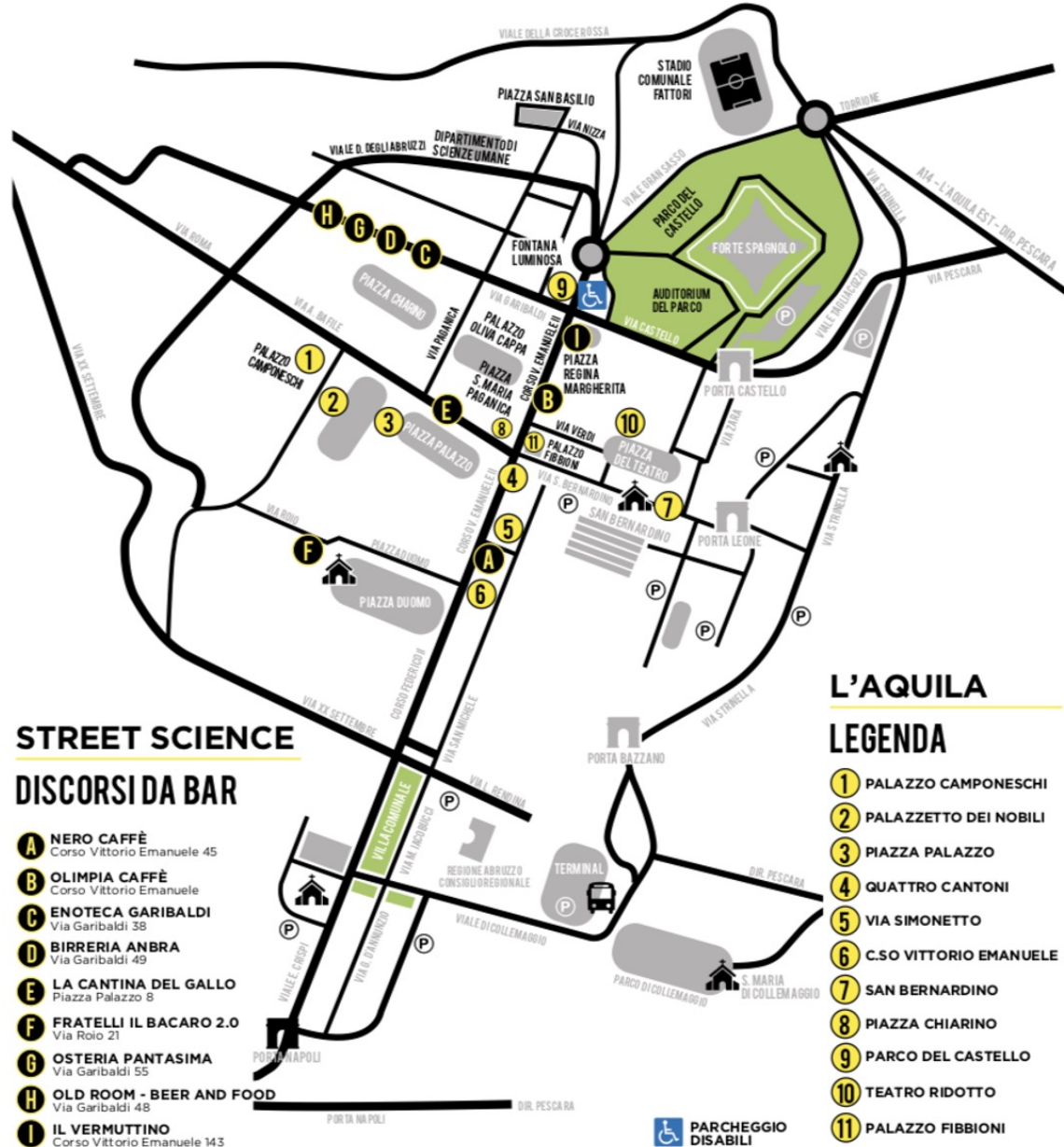
# Architecture Description





Apply IEEE 42010 to NdR  
Case Study

# NdR: European Researchers Night



# The NdR Case

## Key Observations

1. Around 35K visitors per year
2. Late hours are more crowded than early hours
3. Weather affects user's preferences

## Goal

1. A solution for crowd management
2. Automated venue and parking lot management

## Constraints

Limited power and service accuracy constraints





Lets get the requirements  
right



# Functional and Non-functional Requirements

## Functional Requirements

FR1: The app shall allow users to register to the NdR event

FR2: The app shall allow users to provide their preferred events

....

## Extra(Non)-functional Requirements

NFR1: The app must provide 1 second response time or less in web browsers

NFR2: The app should be able to support 1000 users/second while maintaining optimal performance

.....



Who are the stakeholders?

# Stakeholders



Software Developers



IoT Engineers



Urban planning



Software architect(s)

Many more....

# Concerns

Language choice, the  
modules,  
Interactions, ...



Software Developers

Memory,  
Battery,  
platform ...



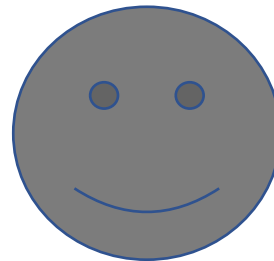
IoT Engineers

Space constraints,  
Location for sensors,



Urban planning

System performance,  
Integration,  
Management,...

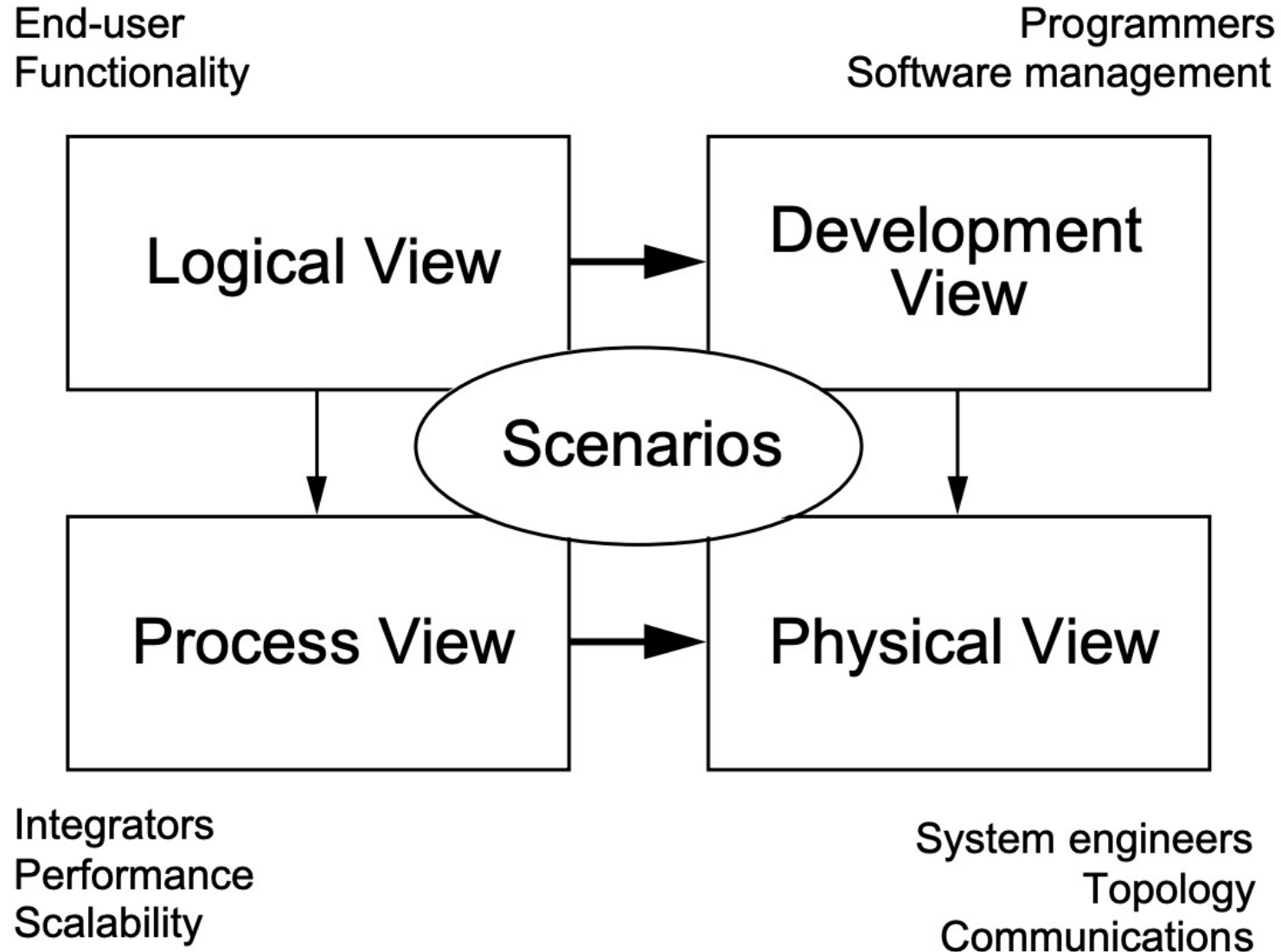


Software architect(s)



What can be the viewpoints  
and views?

# Can we create some models for each view?





What are some scenarios?

# What can be some scenarios?

Book venue

Check  
availability

Check weather

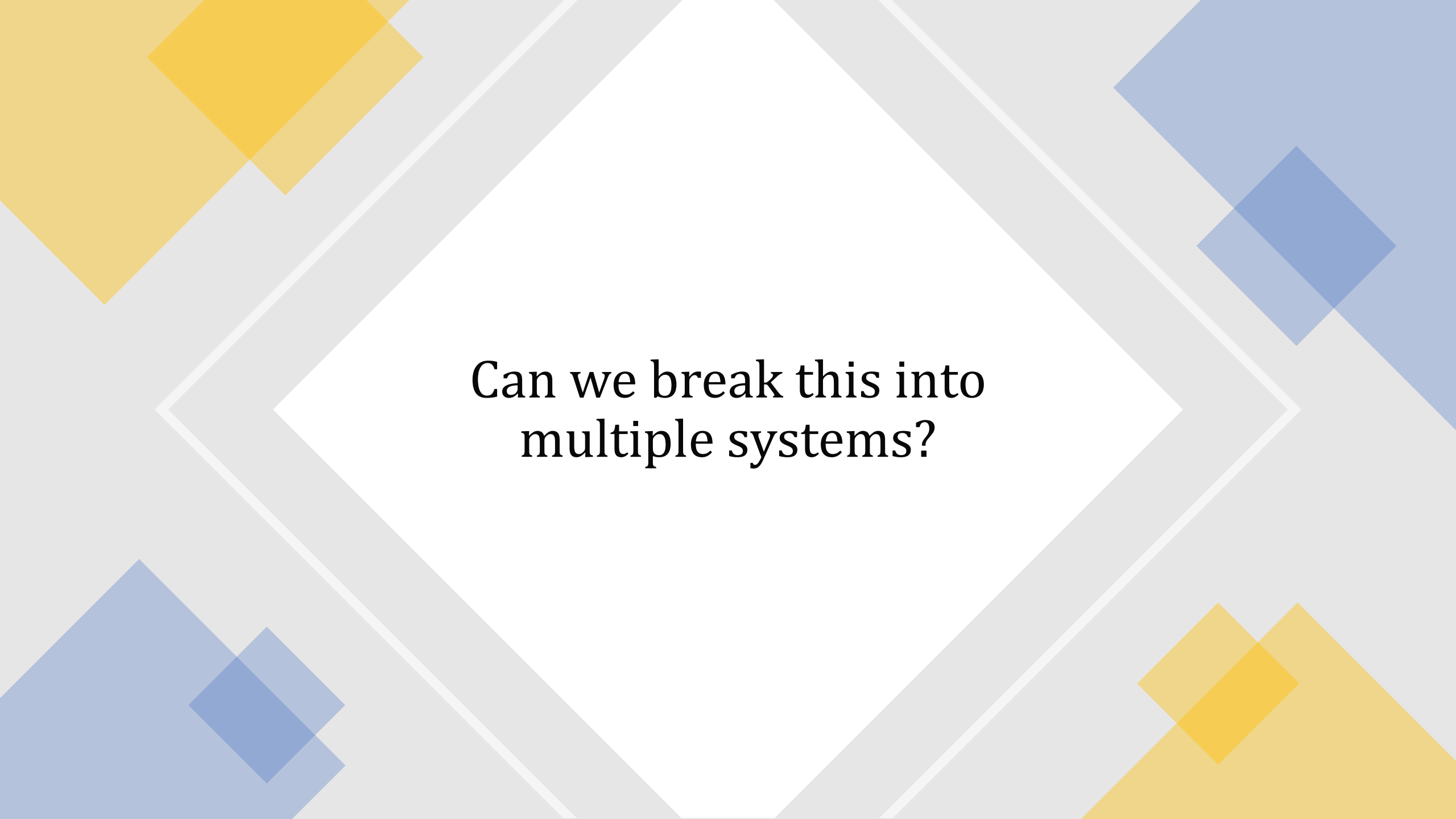
Get  
recommendations

Get confirmation

Book parking lots

Many more...each can be associated to a type of user





Can we break this into  
multiple systems?

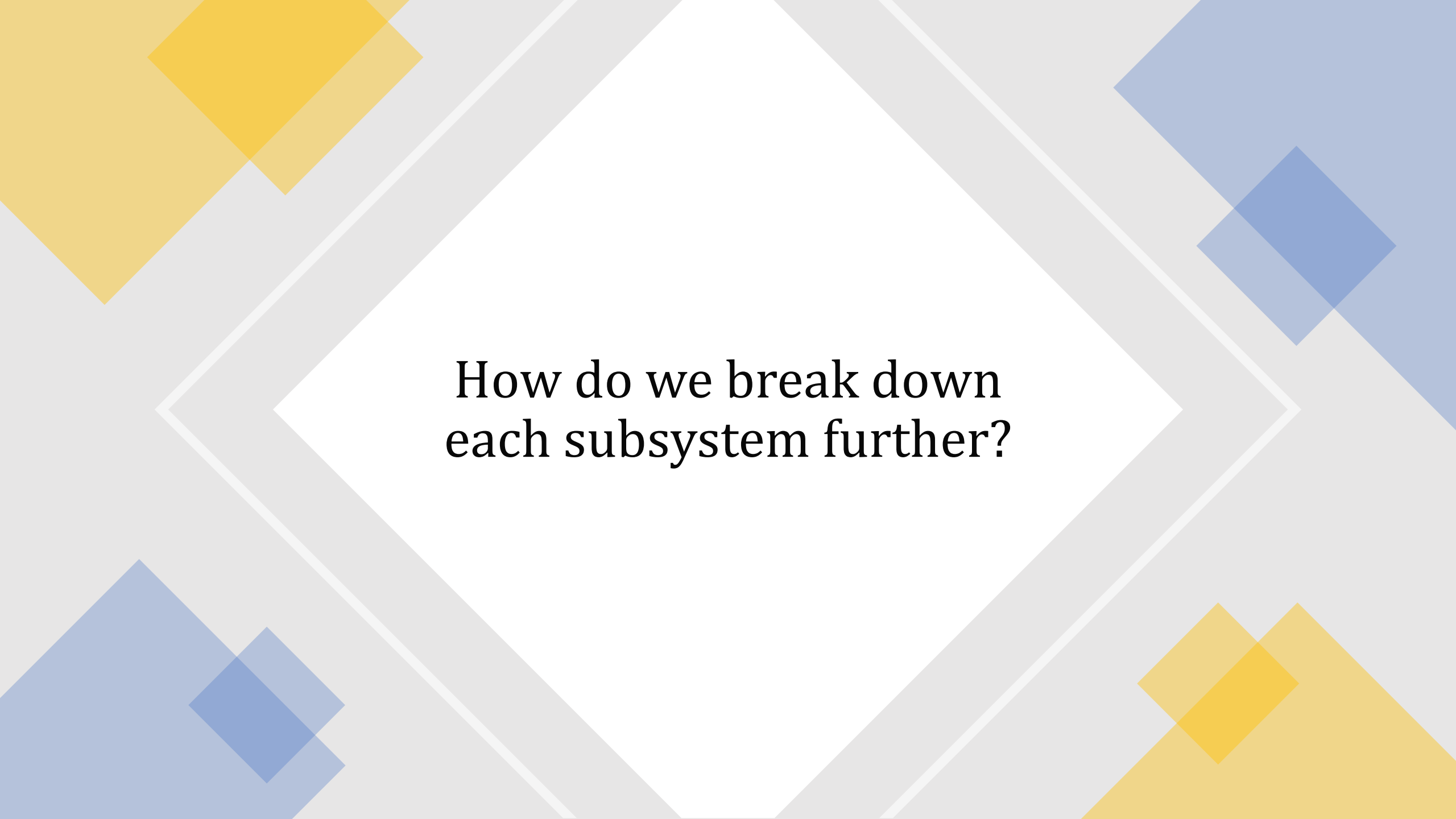
# Subsystems in NdR System

IoT System

Booking System

Weather System

Analytics System



How do we break down  
each subsystem further?

# IoT system

Sensor  
Controller

IoT middleware

Database

Visualization  
Engine

# Booking System

Web Application

Mobile Application

Database

NdR Backend



We can go further

# NdR Backend

Venue Booking  
Manager

Parking Lot  
Booking Manager

Recommendation  
Generator

Payment Manager

And many more..can you name?



Further breakdown? – Think  
of classes...

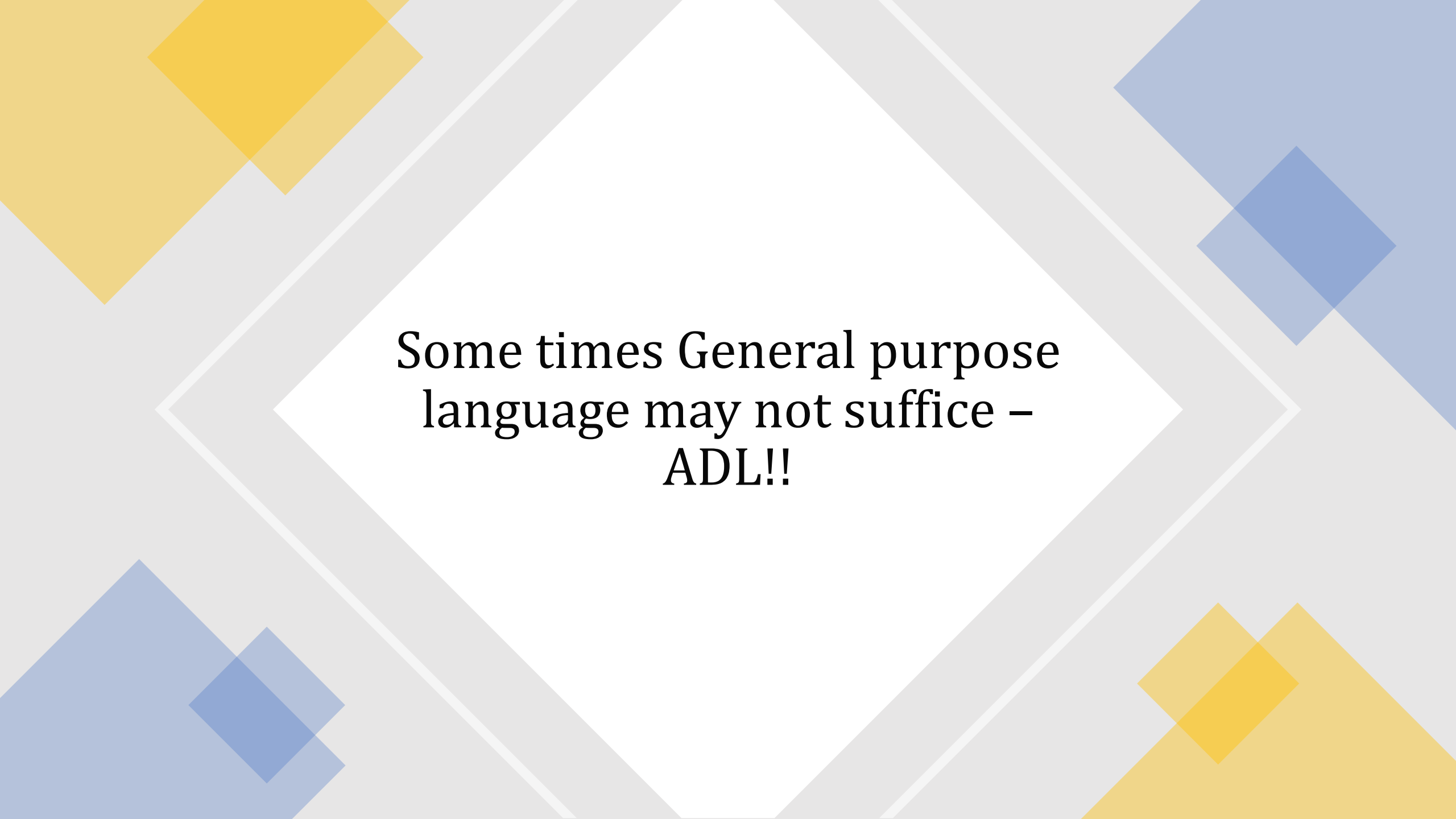




How to model? – UML,  
C4Model,.....

# C4Model

- Created by Simon Brown, Independent consultant (specialization: Software architecture)
- Overcome the challenges of UML, more intuitive language
- Visualize architectures in terms of Context, Containers, Components and Code
- Very developer friendly approach to software architecture diagramming



Some times General purpose  
language may not suffice –  
ADL!!

# Is Software Architecture just some box and arrow?

*“Aside from providing clear and precise documentation, the primary purpose of specifications is to provide **automated analysis** of the document and to expose various kinds of problems that would otherwise go undetected”*

- Perry and Wolf, 1992

*“An architectural system representation is often essential to the **analysis and description** of the high-level properties of a complex system”*

- Garlan and Shaw, 1994

# Architecture Description Languages

An Architecture Description language (ADL) or an Architecture Definition Language is a

- Formal specification language
- For describing the structure and behavior of a software system

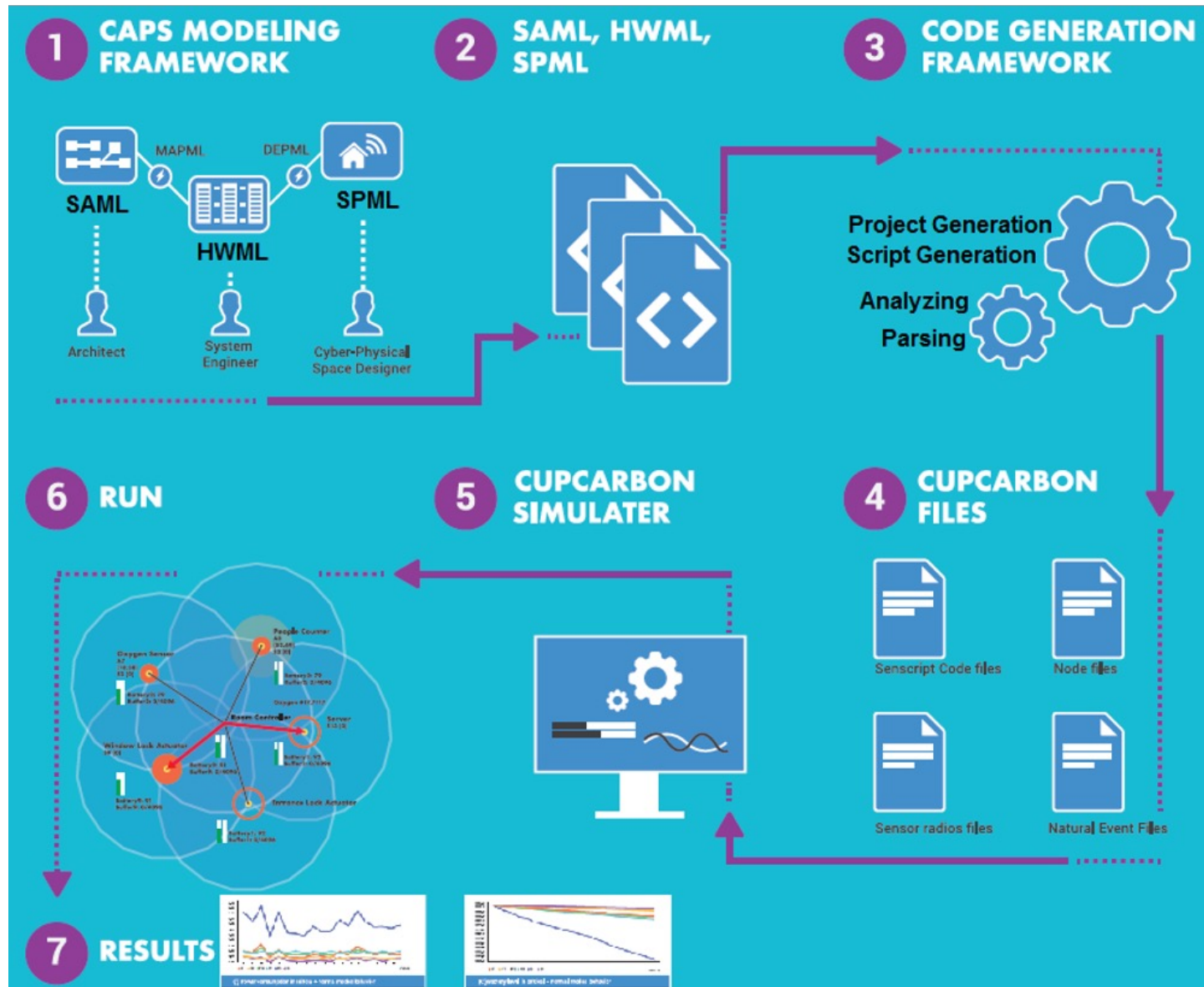
Some popular ADLs: Darwin, ACME, AADL,.....

# Architecture Description Languages - CAPS

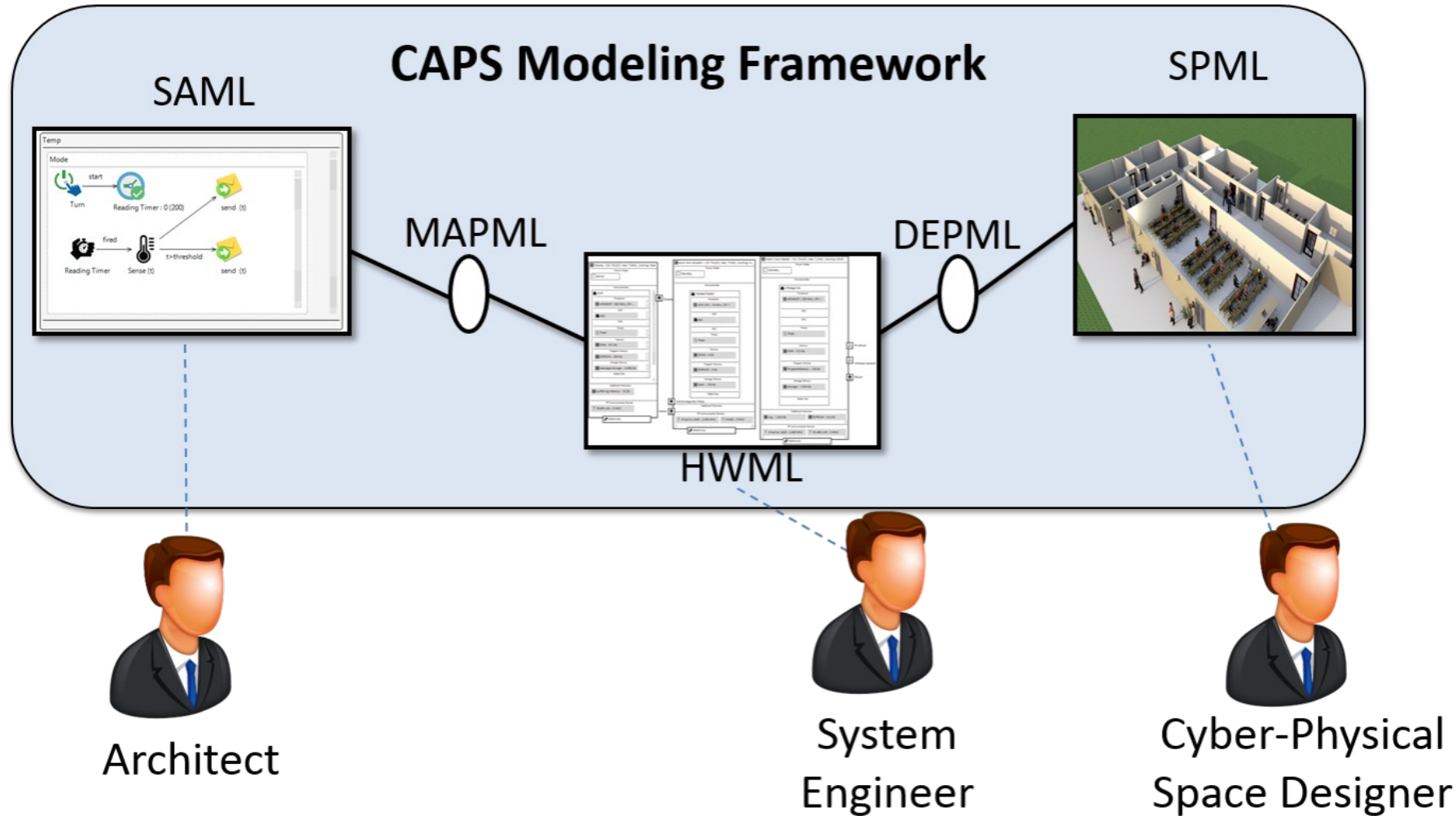


- Evaluate the systems **early** in the design process before the system or prototypes are built [IoT Systems]
- Avoiding **costly** redesign/re-development cycles

# CAPS Modeling Framework



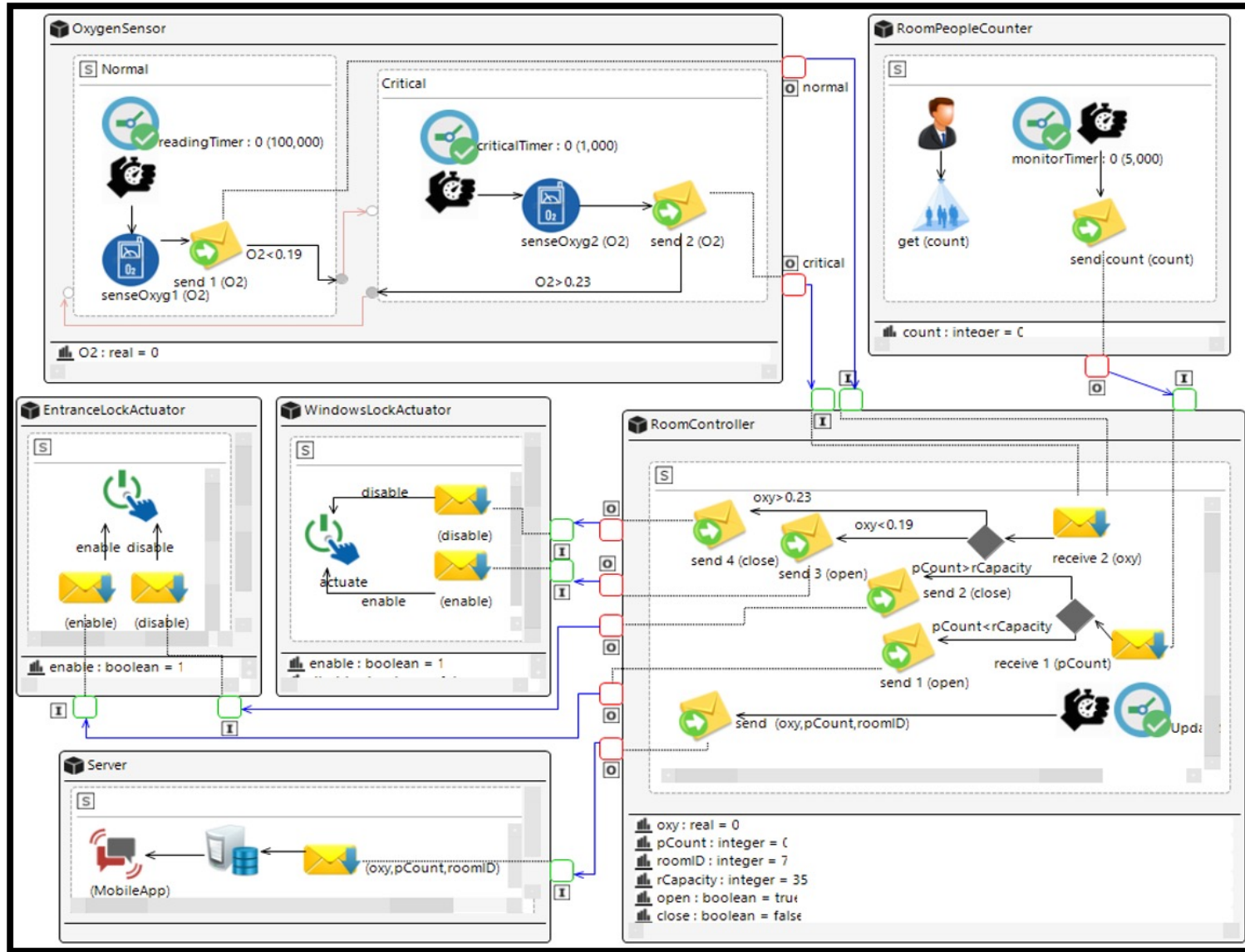
# Multi-view Modeling Framework



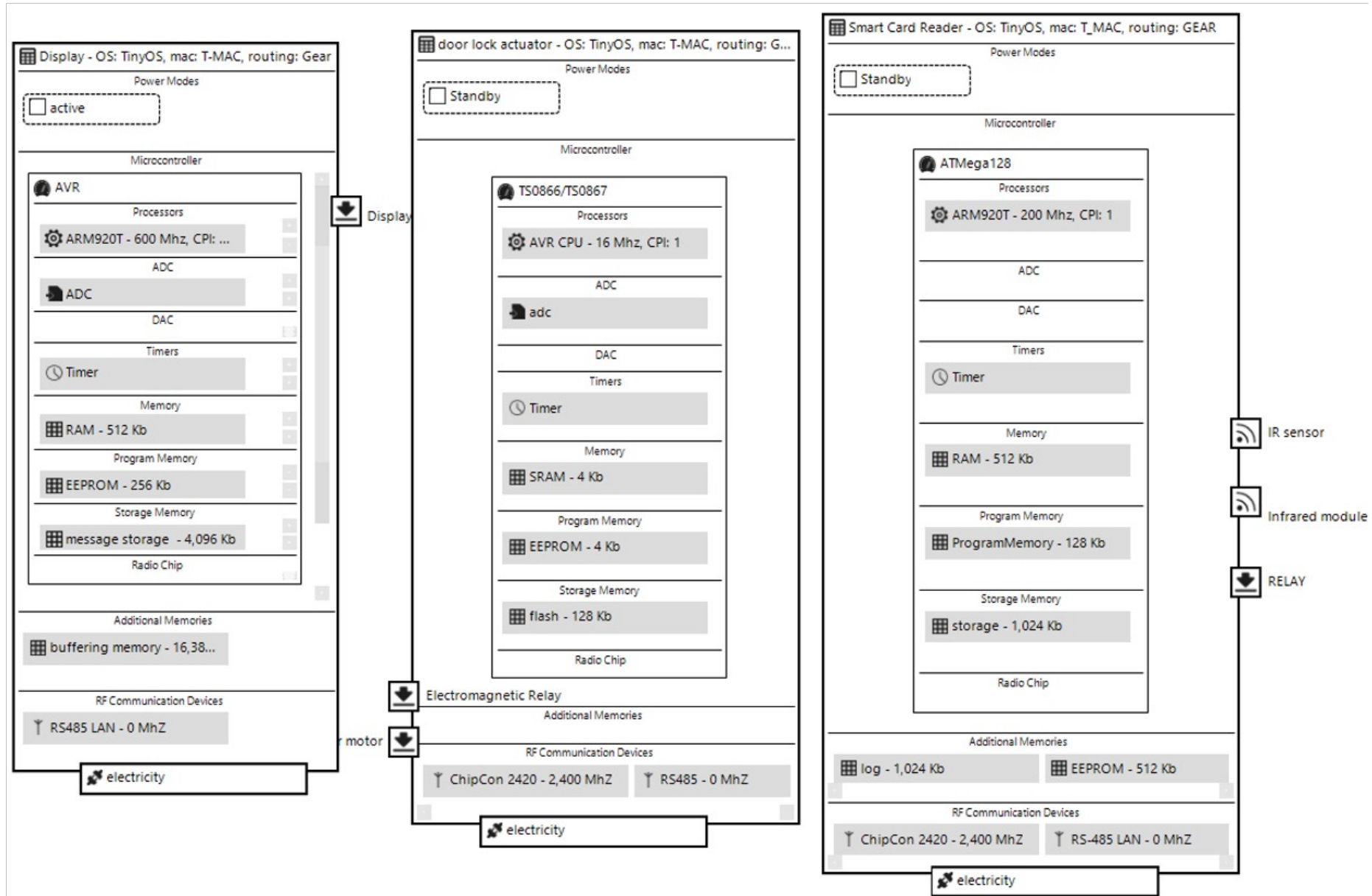
SAML – Software view, HWML – Hardware view, SPML – Space view



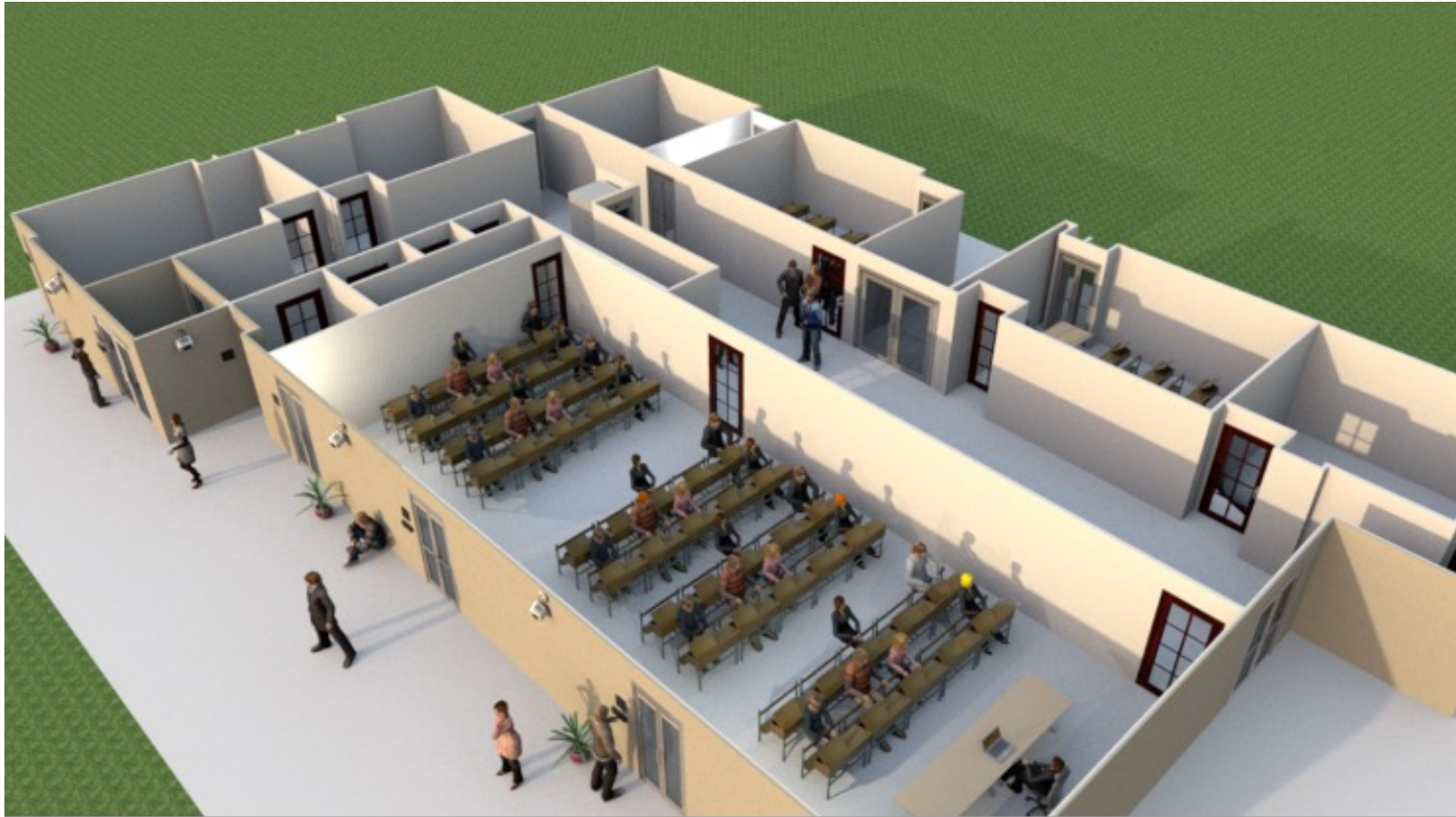
# Software Architecture Modeling Language



# Hardware Modeling Language

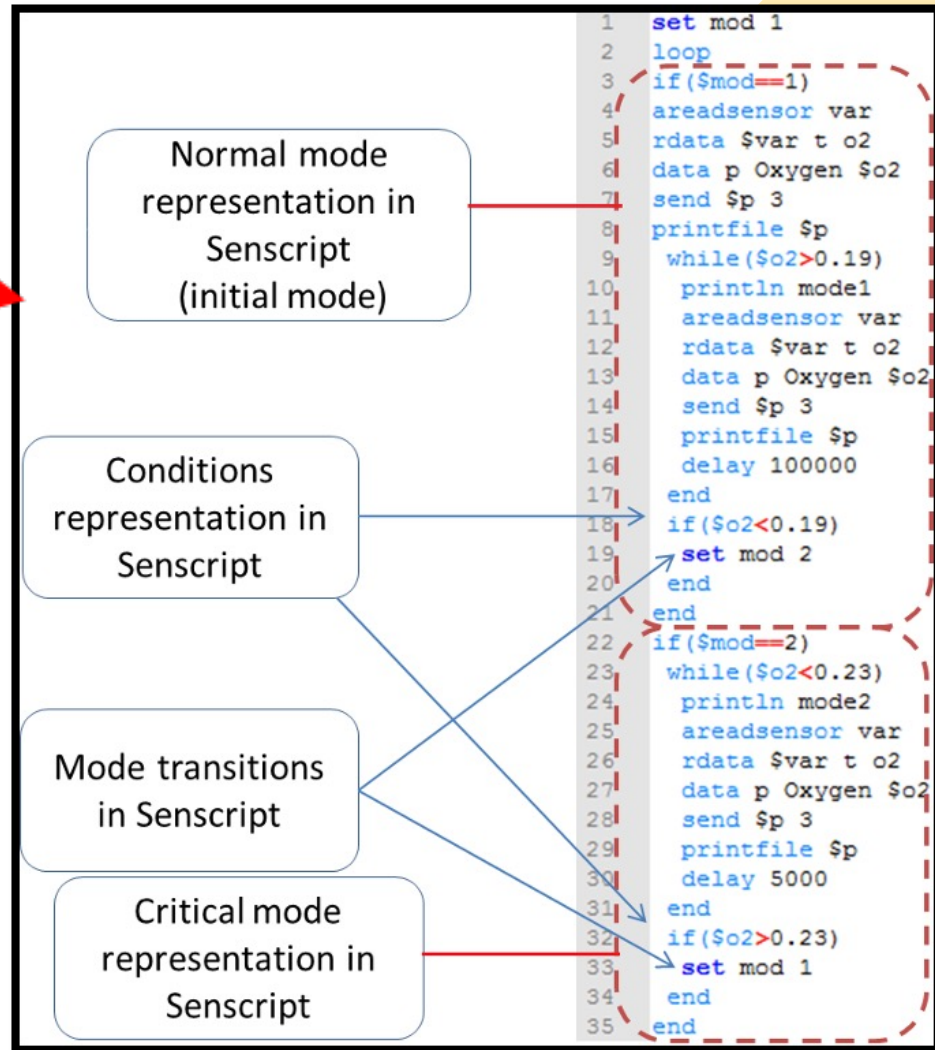
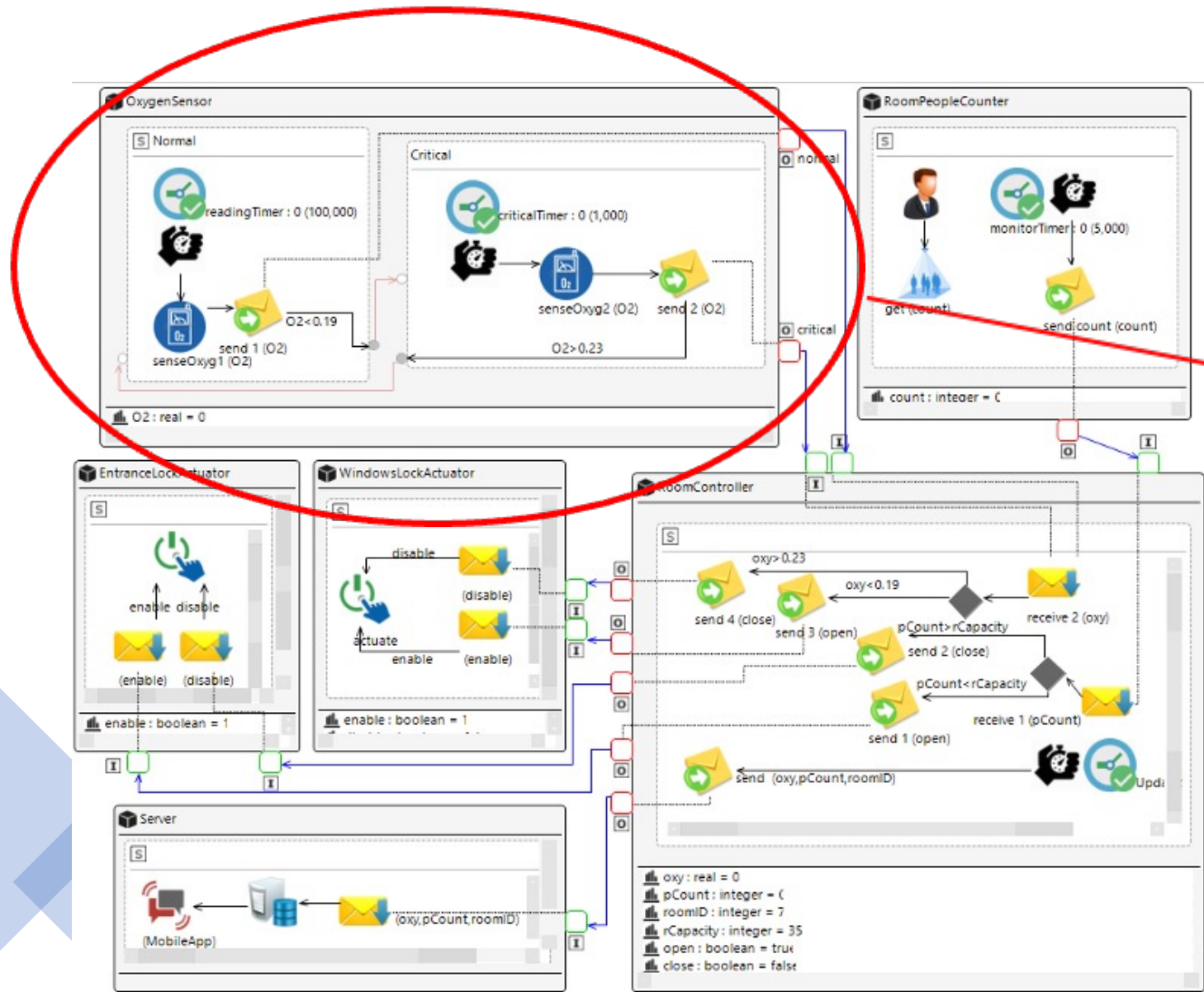


# Space Modeling Language



This will be converted to an xml model

# Space Modeling Language




# Code Generation

The screenshot displays a network simulation environment. On the left, a sidebar contains a tree view with categories: Network information, Devices & Objects, Device Parameters, Radio Parameters, Marker Parameters, and Simulation Parameters. The Simulation Parameters section includes a 'SenScript' button, input fields for Simulation time (86400.0 s), Simulation Speed (1000 ms), and Arrow speed (1000 ms), along with 'Run Simulation', 'Stop Simulation', and 'Apply' buttons. Below these are checkboxes for Symmetrical links, Visibility, Log, Results (0.1 s), Mobility / Events, Clock Drift, ACK, Show, and MAC Layer (1.0).

The main area shows a map with a network topology overlaid. Nodes are represented by icons with labels and status values in brackets. Nodes include: A27, A28, A37, A38, A39, A44, A45, A10, A1, venue3Entrance, venue3Exit, venue2Exit, venue2Entrance, venue1Entrance, venue1SensorExit, parking1Entrance, parking1Exit, parking2Entrance, parking2Exit, compute, database, controllerV3, controllerV2, controllerV1, controllerP2, controllerP1, displayV2, displayV1, displayP1, displayP2, S24, S25, S26, S34, S35, S33, S42, S43, S44, S45, S46, S47, S48, S49, S7, S11, S17, S1, S2, S3, S4, S5, S6, S8, S9, S10, S12, S13, S14, S15, S16, S18, S19, S20, S21, S22, S23, S27, S28, S29, S30, S31, S32, S36, S37, S38, S39, S40, S41, S43, S44, S45, S46, S47, S48, S49, S50, S51, S52, S53, S54, S55, S56, S57, S58, S59, S60, S61, S62, S63, S64, S65, S66, S67, S68, S69, S70, S71, S72, S73, S74, S75, S76, S77, S78, S79, S80, S81, S82, S83, S84, S85, S86, S87, S88, S89, S90, S91, S92, S93, S94, S95, S96, S97, S98, S99, S100. Each node has a status value in brackets, such as [0.00] or [0].

At the top right of the map area, a console window displays the following text:  
Time: 0.0000 s  
Number of SENT messages:0.0 [0.0]  
Number of RECEIVED messages:0.0 [0.0]  
Number of ACK messages:0.0 [0.0]  
Number of LOST messages:0.0 [0.0]

The bottom of the interface features a toolbar with navigation and view controls, including buttons for 'Connections', 'Nodes', 'Radios', 'Sensors', 'All', 'Details', 'Distances', 'Links', 'Node Arrows', 'Map Contrast', 'Progression', 'N=32', and 'RT'.



How do you guarantee  
quality – Can we think of  
tactics?

# Thank You



Course website: [karthikv1392.github.io/cs6401\\_se](https://karthikv1392.github.io/cs6401_se)

Email: [karthik.vaidhyanathan@iiit.ac.in](mailto:karthik.vaidhyanathan@iiit.ac.in)

Web: <https://karthikvaidhyanathan.com>

Twitter: @karthi\_ishere

