MusQraTT

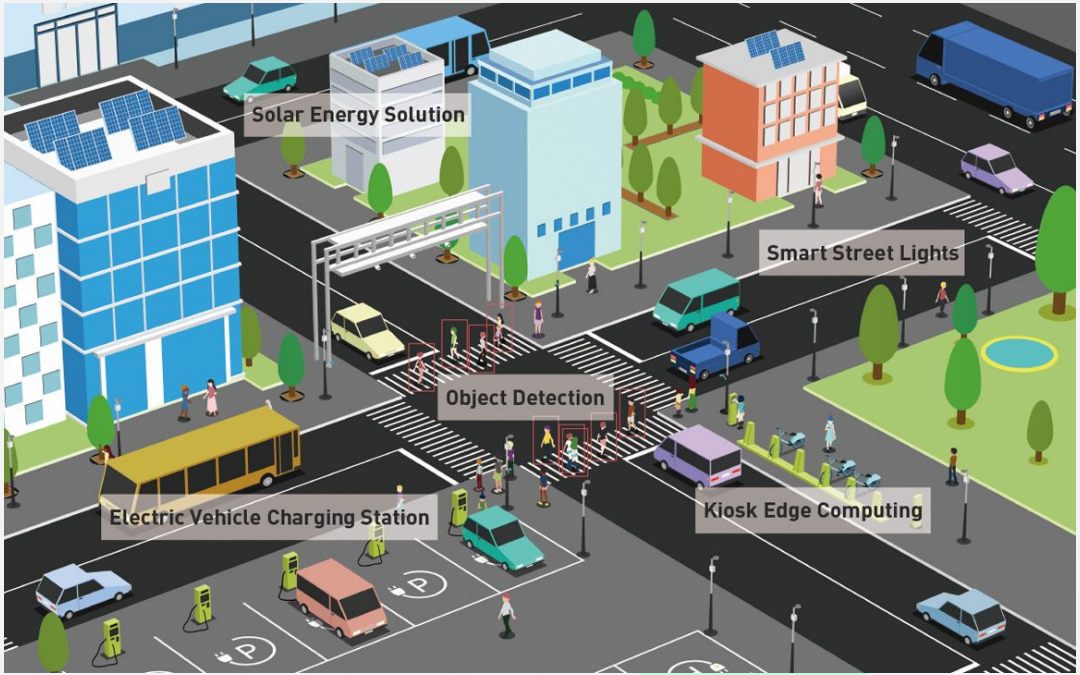
MQTT-SN Broker for the Edge

Aisha Mohammed
Evan Stella

Mentored by:
Prof. Gabe Parmer
Gregor Peach

The Future: “Internet of Things” and Smart Cities

- Our infrastructure is becoming increasingly interconnected and sensitive to communication **delay**, or **latency**
- Time-sensitive devices need a way to communicate near-instantaneously with each other



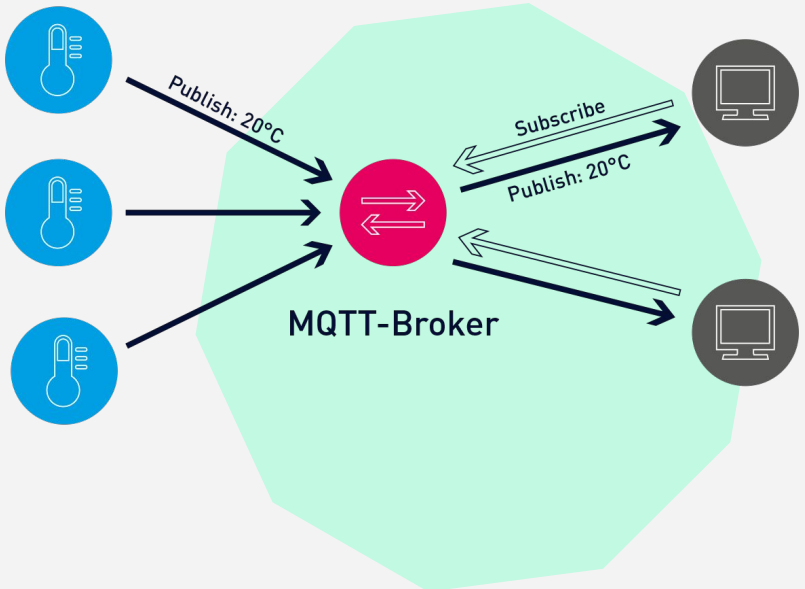
AxiomTech

MQTT-SN

- Lightweight version of MQTT: a Publish-Subscribe network protocol
- Designed for device communication for Internet of Things & cyber-physical systems
    - Smart Homes
    - Industrial Systems
    - Sensor networks

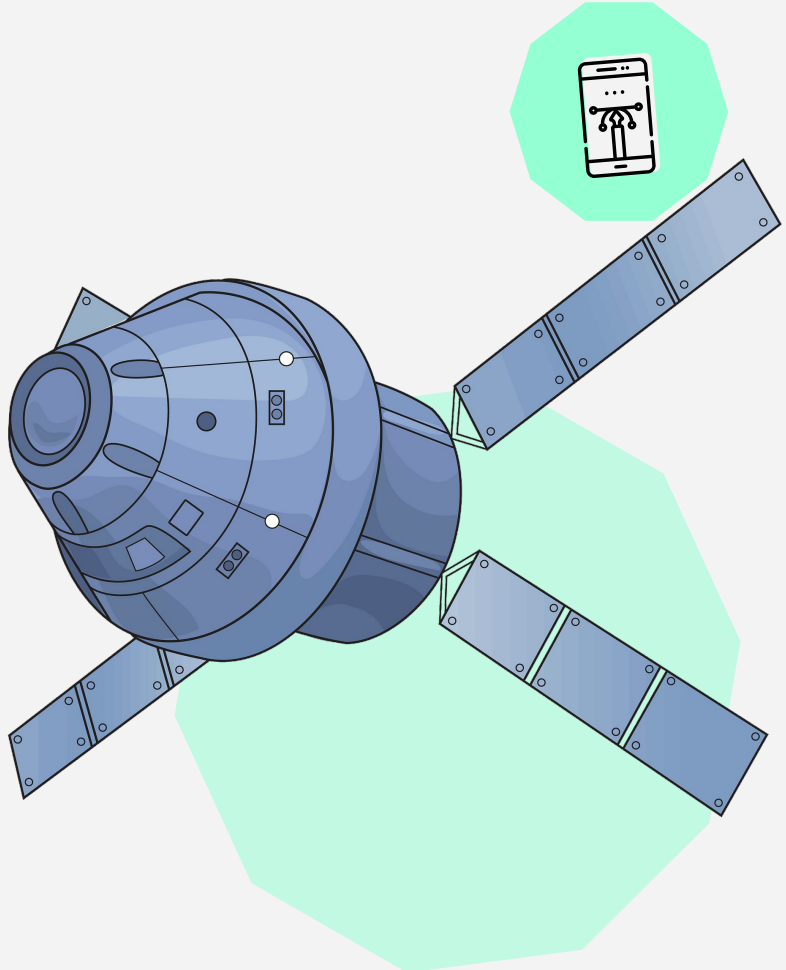
MQTT Broker

- Facilitates communication between clients
- Multiple clients have a connection to the broker
- Clients publish & subscribe to topics managed by the broker
- The Broker distributes data to the subscribers

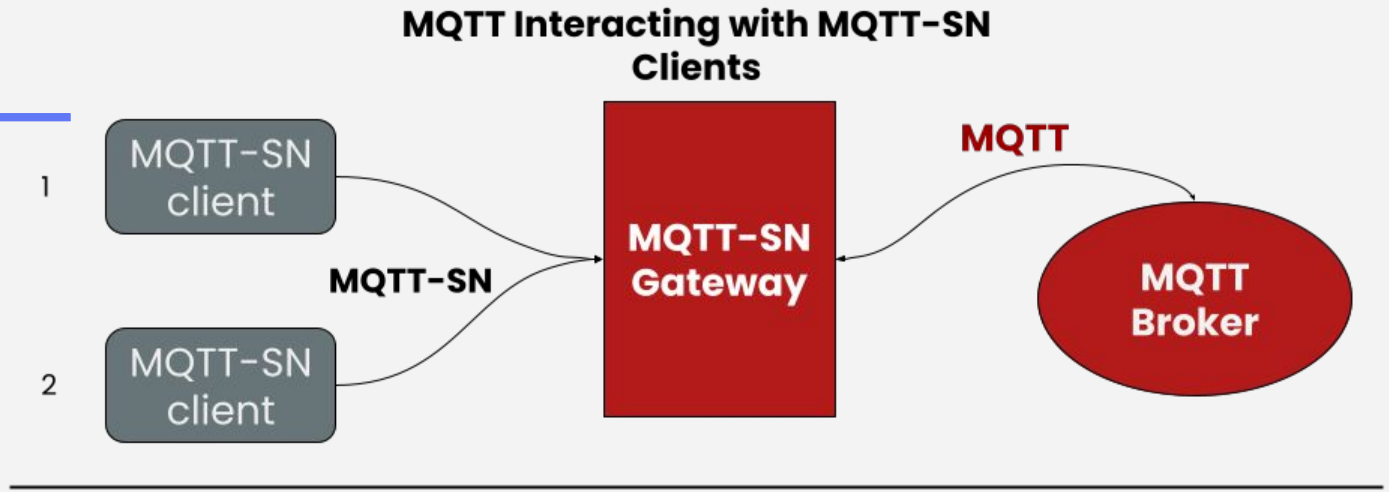


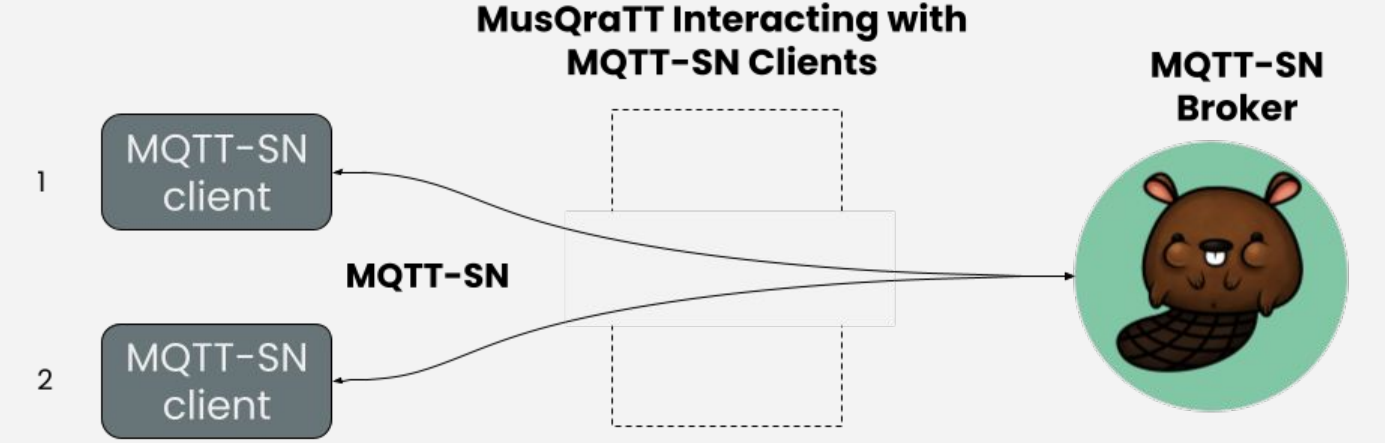
The Current Gap

- The most popular MQTT Brokers available have two problems
  - High transmission latency
  - Heavyweight runtime environment
- This makes them unsuitable for:
  - Autonomous vehicles
  - Spacecraft
  - Critical/latency sensitive infrastructure



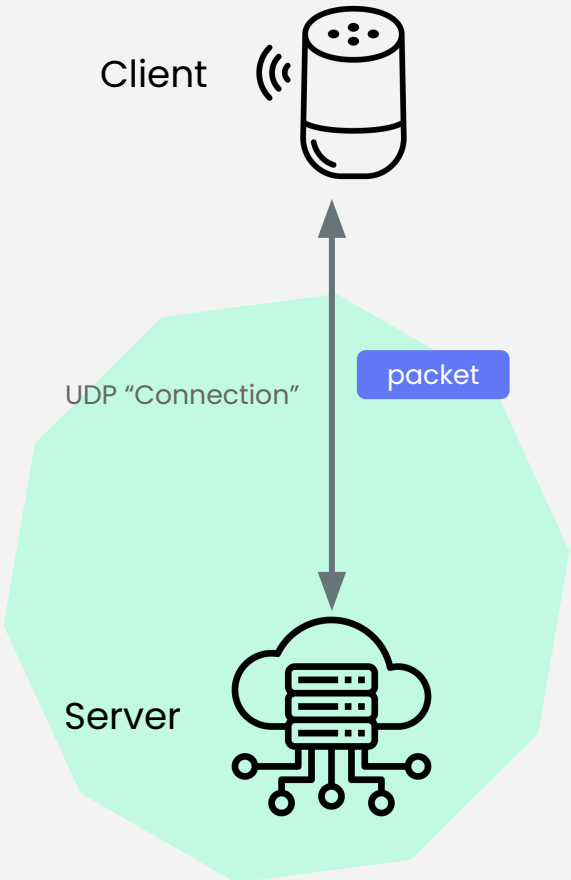
Solution


MQTT Interacting with MQTT-SN Clients
1
MQTT-SN client
2
MQTT-SN client
MQTT-SN
MQTT-SN Gateway
MQTT
MQTT Broker


MusQraTT Interacting with MQTT-SN Clients
1
MQTT-SN client
2
MQTT-SN client
MQTT-SN
MQTT-SN Broker

How does it work?

- **The client and server (broker) send encoded packets over UDP**
- By subscribing to a topic, the client receives published data as it becomes available
- When data is no longer needed, clients can unsubscribe
- Maintains a list of all clients, and their subscription to topics



MusQraTT

The Rust Programming Language

\+

CompositeOS

MQTT-SN Broker written in Rust

- Rust provides enhanced memory management and low-level concurrency
- Lightweight runtimes makes it optimal for Edge, real-time devices

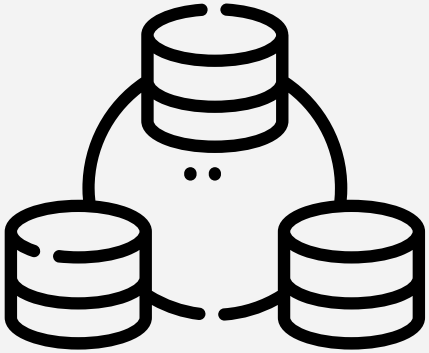
The Rust Programming Language

MQTT-SN Broker on CompositeOS

- Operating system developed by GW
- Created to operate on the Edge
- Prioritizes low latency & reliability

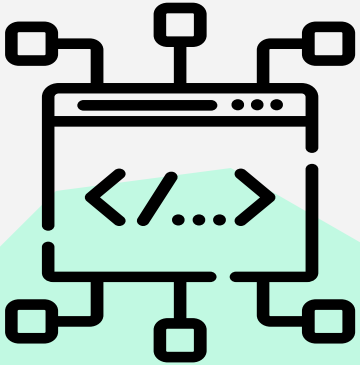
GW
SYSTEMS LAB

MusQraTT's Impact

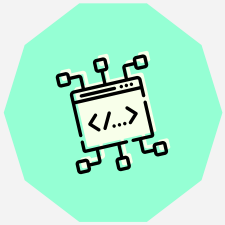


An MQTT-SN Broker for real-time and embedded devices at the Edge

Enables edge computing to provide increased scalability & performance



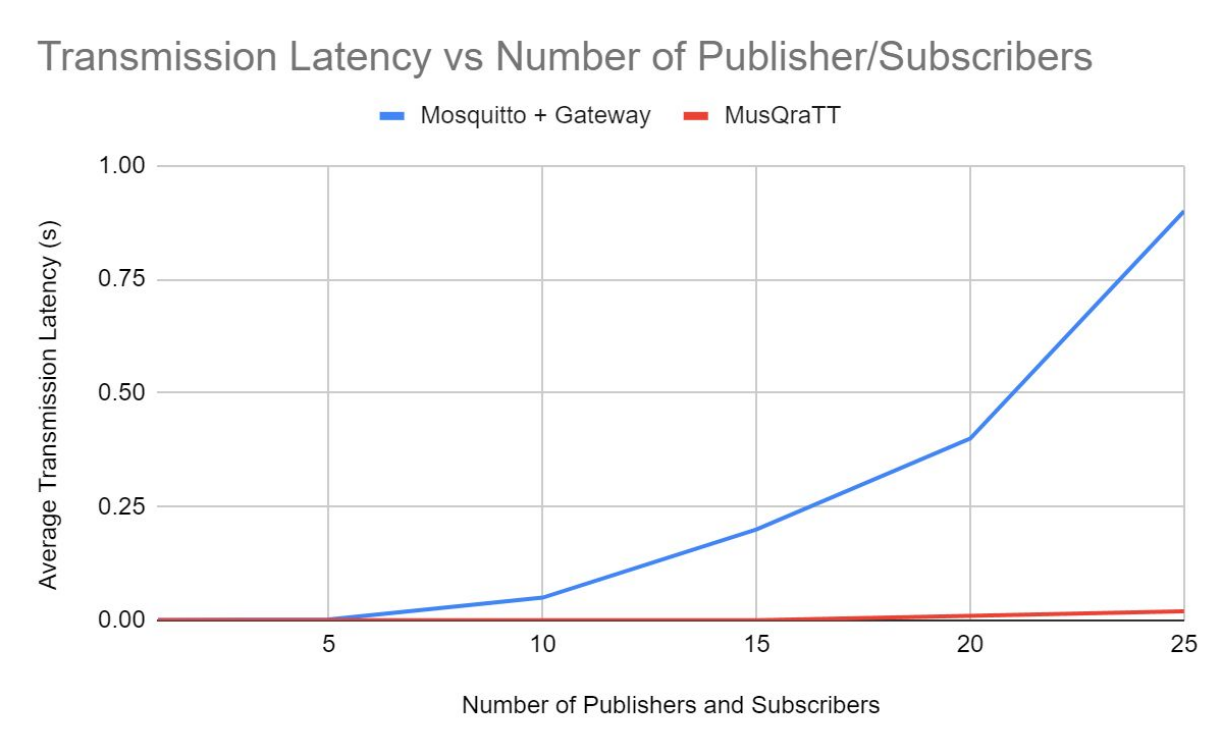
Utilizes Rust & Composite to create a broker suitable for real-time and latency sensitive applications



Results

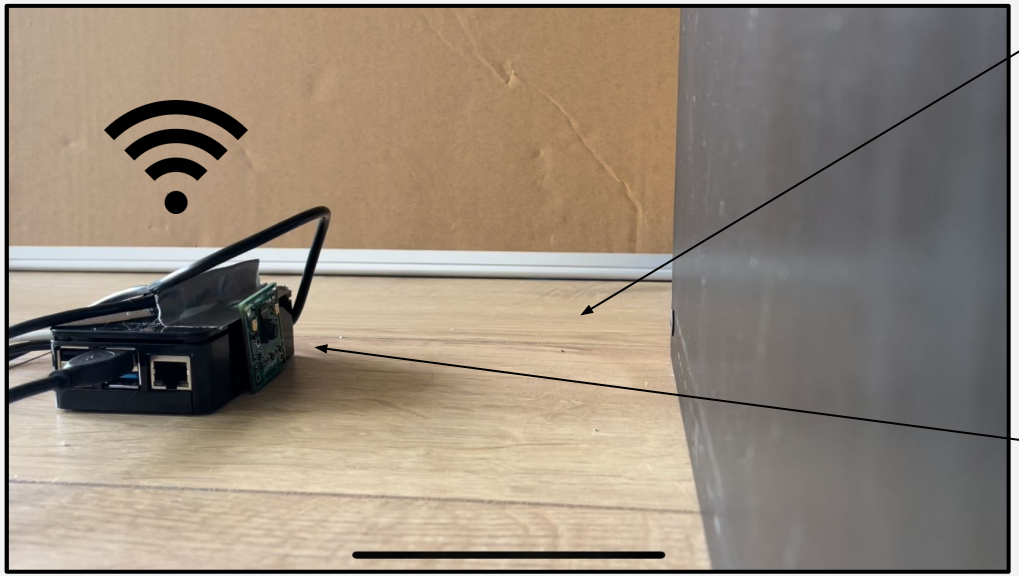
How fast is MusQraTT?

MusQraTT is FAST



Consequences of Latency

Simple Collision Avoidance System



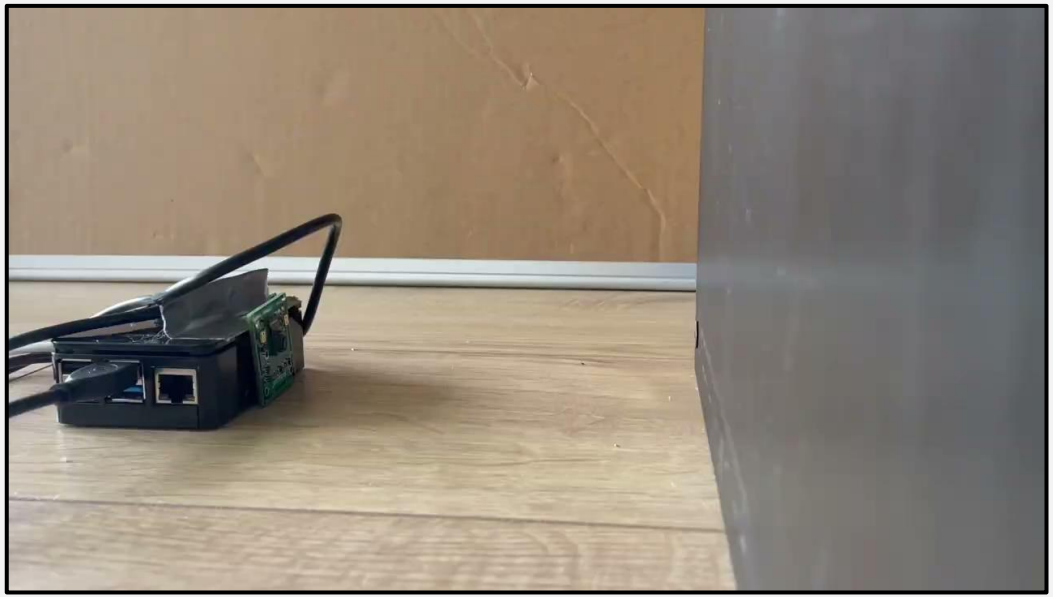
City Intersection

Subscribers: Autonomous Vehicles

Publisher: Collision Detection System

Consequences of Latency

Using **Mosquitto** as the Broker

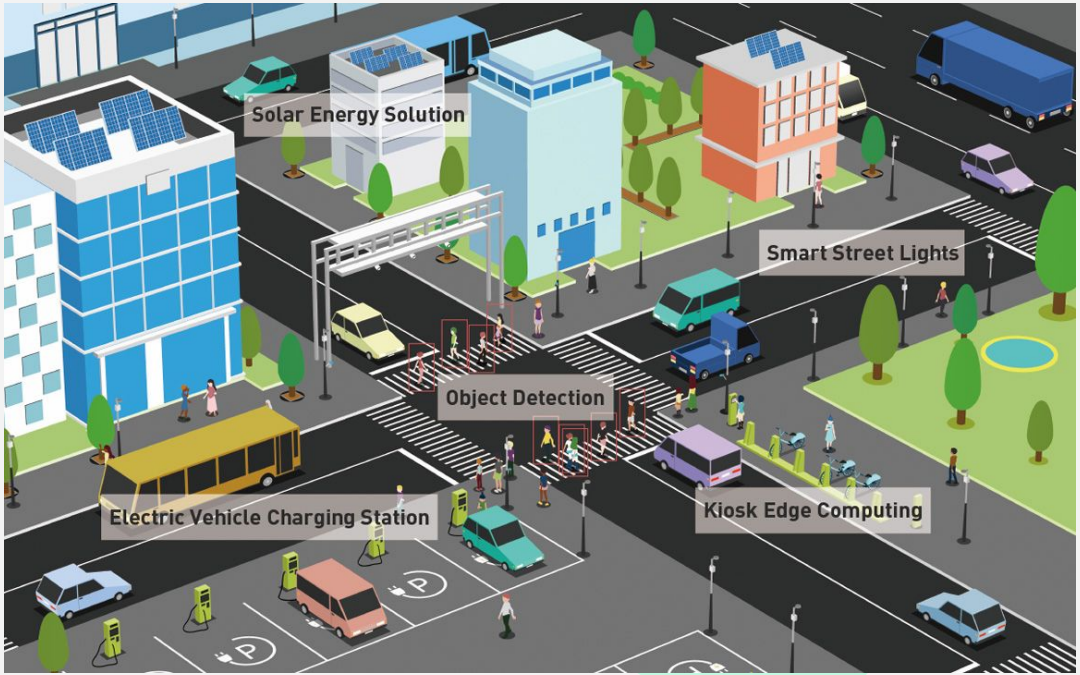


Consequences of Latency

Using **MusQraTT** as the Broker

The Future: “Internet of Things” and Smart Cities

This research is critical to enabling the low-latency systems that will have a high-impact on our future



AxiomTech

Thank you!

Special Thanks to Dr. Gabriel Parmer, Gregor Peach, and the course instructors!