FIWARE IoT Platform for Agricultural Robot Fleet Management

23/02/2024 VTT – beyond the obvious

Contents

- Intro
- Problem
- Proposed solution
  - System design
  - Components
- Piloting
- Experiences
- Conclusions

Introduction

- VTT has been involved for a long time with FIWARE
- FIWARE has been deployed and piloted in various EU project pilots
  - Smart cities and buildings
  - Agriculture
  - Sensor networks
- Standard data model (NGSI) is the heart of the FIWARE
- Most common Generic Enablers
  - Context Broker, Timeseries Database
  - Other components as needed

Agricultural Robotics and FIWARE

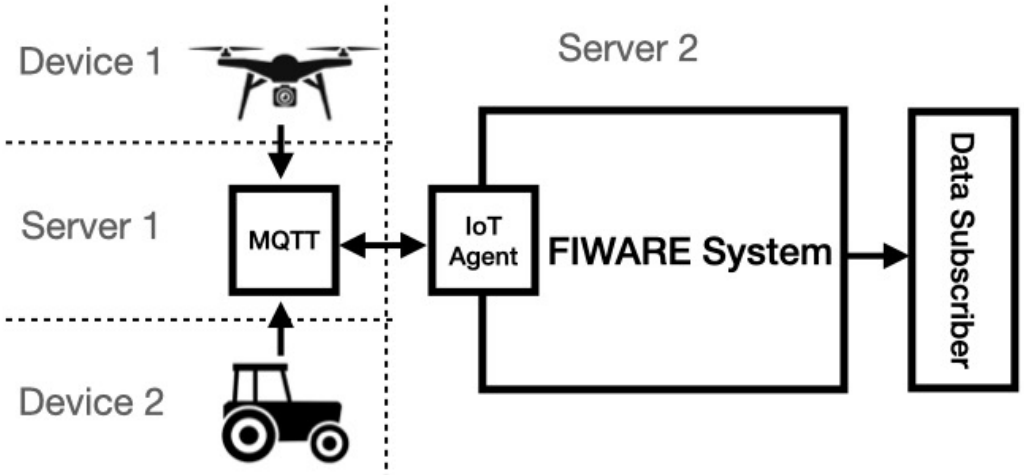
- Use case in FlexiGroBots EU Project
  - The use of heterogenous fleets of robots to perform agricultural tasks
  - Multiple operators, multiple robotic platforms
  - No integration of robots on the robot system level
- Goal
  - To enable robots to work on the same field performing their tasks
  - With a minimal operator workforce
  - Additional safety layers

Proposed Solution

- Introduce minimal integration of different robot systems
- Basic set of functions (Start, Stop, Pause, Resume, Status)
  - Geared towards safety
- Simple adapters for each robotic system consuming and producing data in common format -> NGSI

System Design

- Adapter for robot control system
  - For instance DJI Mobile SDK
- FIWARE platform deployed on a cloud server in Docker environment
  - Situational Awareness service as well
- Desktop application for fleet operators to monitor all robots during operation
  - Uses MQTT to receive statuses and events

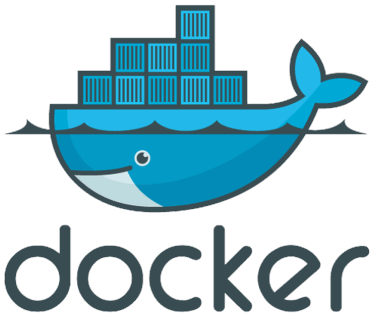


Components

- Robot control system

- MQTT Broker
- IoT Agent
- Orion
- MongoDB
- QuantumLeap
- TimescaleDB
- SituationalAwareness Service

- FleetManager



Piloting

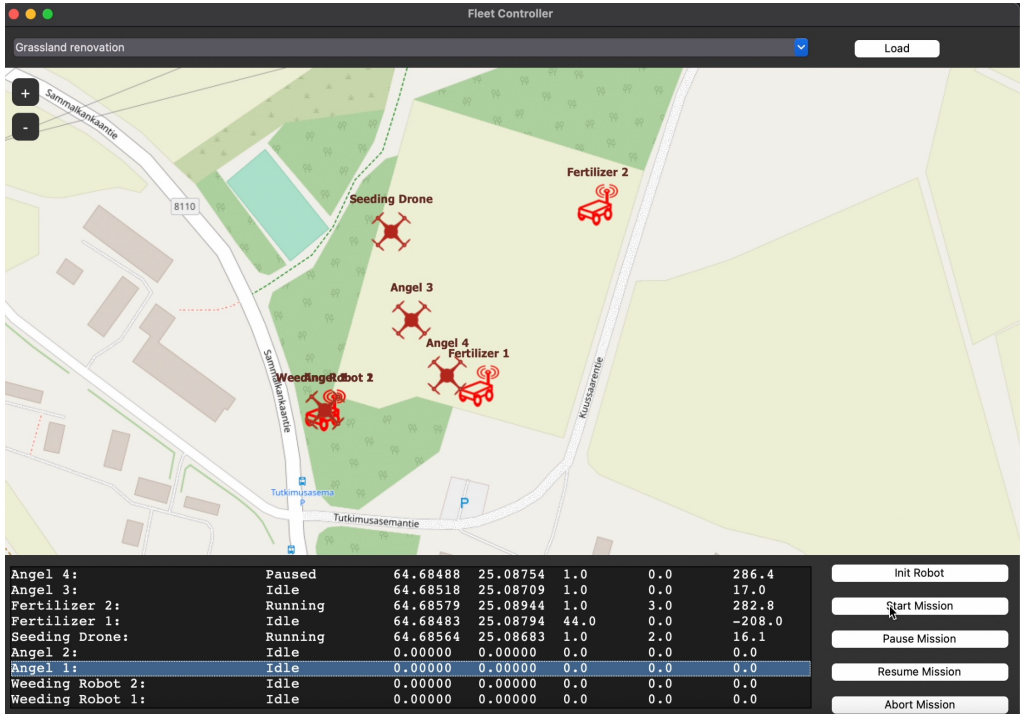
- Drones
  - Survey drones, Surveillance drone, Spraying Drone
- Ground Robots
  - Weeding COBOT, Robot gantry, Tractor, Seeding robot, Collector robot
- Connectivity via 4G/5G
- Services running in private cloud

Data Model

- Simple Two entity type data model
  - NGSI v2 because NGSI-LD broker was still not at the beginning of the project
- Robot –entity
  - ID
  - Location
  - Speed
  - Status
  - Heading
  - Time
- Collision
  - Location
  - Array of Robots involved
  - Time to Collision
  - Timestamp

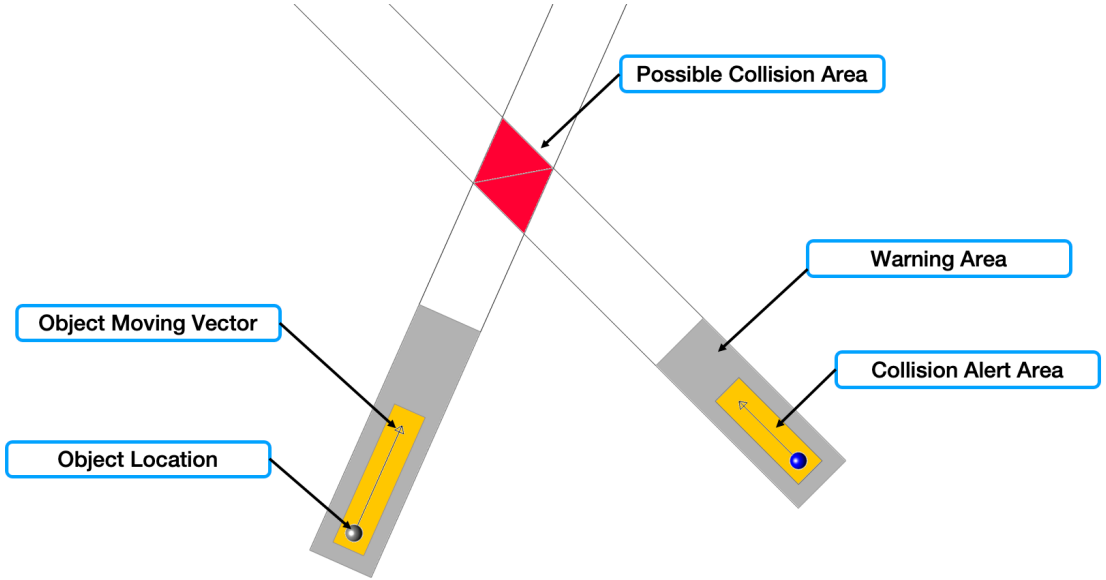
Fleet Management

- Monitor & Control robots and drones
- Connected to FIWARE via MQTT broker and IoT Agent
- Also connected to data space

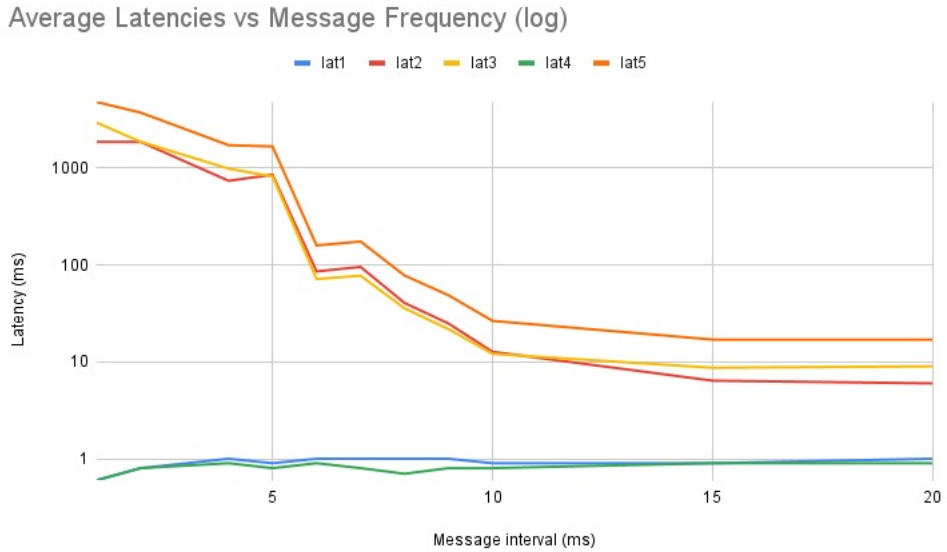


Collision Avoidance

- FIWARE application subscribing to robot statuses and calculating possible collision events

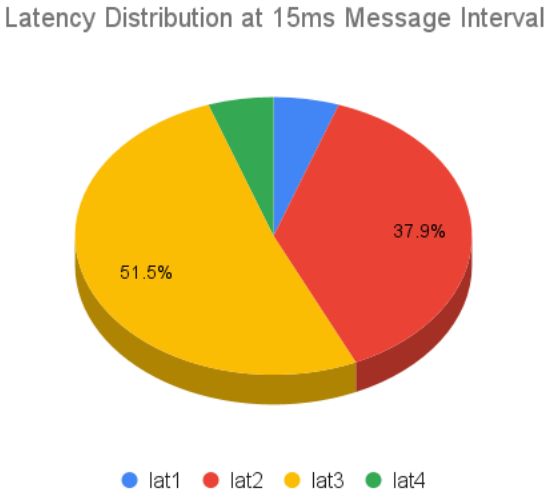


Experiences



- Critical factor is latency
  - Latency affects the safety margins
- With close to 10 robots, the system is still useable

Experiences



- IoT Agent and Orion are responsible of most of the latencies
  - MQTT UltraLight to NGSI IoT agent consumes 37.9% of the time (lat2)
  - Orion context broker consumer 51.5% of the time (lat2)

Conclusions

- Platform build using FIWARE components is relatively easy to deploy and use
- Standard data models help in the integration
- The downside of the standard data model is the performance
  - Data conversion is heavy compared to native binary protocols
  - ➢ This has to be taken into account in system design

Thank You!

- Question & Answers

- Kari Kolehmainen
- kari.kolehmainen@vtt.fi
- +358 40 5330649