Mediator system for robotic applications

Rubanraj Ravichandran

Master Thesis Proposal
Master of Autonomous Systems

Department of Computer Science
University of Applied Sciences Bonn-Rhein-Sieg

Supervisors:
Prof. Dr. Erwin Prassler
Advisors:
Nico Huebel Sebastian Blumenthal

November 5, 2018

1 Introduction

Robots generate large amount of data from different types of sensors attached to it and also from its hardware components. In our previous research work, we have conducted an extensive qualitative and quantitative analysis to find better databases and architectures that effectively store these data and consume it for further operations. Results from our previous work shows that, a single database is not suitable for every robotic scenarios. For example, in terms handling large BLOB data, mongoDB stored them faster but reading the data was slower compared to couchdb. Also, to complete a given task robots depends on multiple source of information from internal sensors, as well as external sources for example world model, kinematic model, etc..

Adoption of multiple databases for robotic applications requires unique way of mediation to view multiple databases as a single federated database. Mediator approach helps to integrate data from different sources and produce a single result back to robots. Mediator abstracts the information of how data is being stored in different data sources from robot, and it allows robotic applications stream data to mediator independent of databases used in the back-end.

To Map the data generated by robots with multiple databases, mediator system requires a proper data model predefined in the context of robotic applications. Modeling robot generated data helps to generalize the structure of data and defining relations between different objects in the robot systems. If we have a well defined robotic data models, then mediator will get the ability to mutate or query data from different data sources. Also it is important that, these data models should be extended to any robotic use-cases.

As mentioned in these papers[1],[2],[3], mediators are being used to integrate data from different data sources and few architectures supports single data model (e.g SQL), and others supports for different data models (e.g SQL,NoSQL, document store, etc..). Also, they are differ from query languages, ease of implementation, components used in their architecture. This project focuses on defining suitable data models for robotic applications and implementing a mediator system which act as a middle-ware between robots and databases.

2 State of the art analysis

3 Problem Formulation

Our previous work results reveals not all databases reacts in a similar fashion for different heterogeneous data from robotic applications. Also, there are no data models has been defined in the context of robotic application sensor data. For example, black box designed for ROPOD project simulation uses mongoDB to store raw sensor data with out any data model.

(contact alex or santhosh to see the current implementation and complete the problem here)

In terms of supporting multiple databases setup for robots, we need a systematic approach to store and retrieve data from external sources and, a well defined data model that can map components and sensors to a robot and also with the world model.

These are the major problems will be discussed and addressed in this work,

- Data model for robotic applications
- Implementation of single mediation layer to connect between robots and databases (different data model)
- Unique query language that can be used by robots to talk with different databases via mediator.

3.1 Use cases

Need to be defined

4 Workplan

This workplan shows the major decomposition of the workpackages, start time and end time. This project starts on August 15th, 2018 for 6 months duration and ends on February 15th,2018.

4.1 Work packages

5 Deliverables

5.1 Minimum

- Report on state of the art analysis.
- Collection of existing IoT device management tools which are capable to do OTA updates.
- Qualitative comparison of these tools based on the work flow, advantages and disadvantages.

5.2 Expected

- Designing and implementing test bed for peer-peer and multicast protocol.
- Running experiments and collection of results.
- Analysis of results.

5.3 Maximum

- Incorporate best performed decentralized file sharing protocol with CP-Swarm platform to handle OTA update.