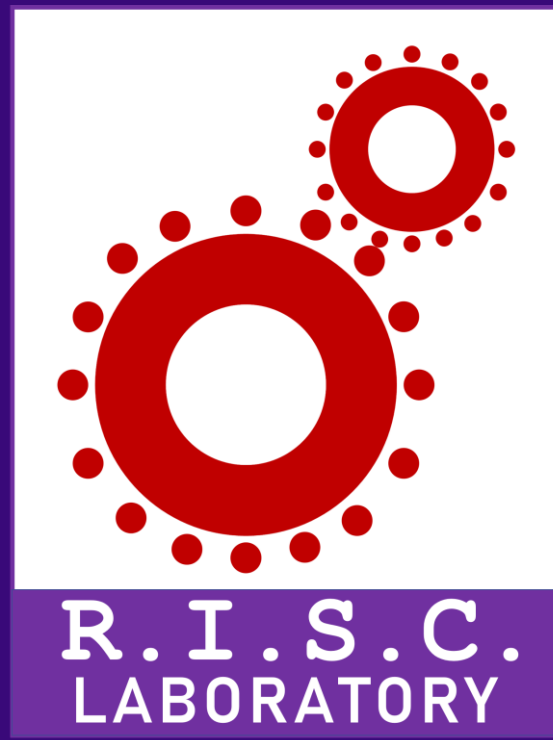


# MQTT and ROC Based Hybrid Robot as a Service (RaaS) Platform

Pritesh C. Bhavsar, Prof. Sarosh Patel, Dr. Tarek Sobh  
 Department of Computer Science, University of Bridgeport,  
 Bridgeport, Connecticut, USA



## Abstract

Robots are rapidly evolving from factory work-horses to robot-companions. The future of robots will be as companions in the workplace functioning as interactive salespeople. In order to support this transition, it is important to combine service-oriented architecture and robotics. Service-oriented architecture and cloud computing have become dominant computing paradigms, and adding an RaaS (Robot as a Service) unit as a part of this system will help the companies manage and develop robots more efficiently. The major components of RaaS will be the integration of RMS (Robot Management System) and ROC (Robot Operation Center). As more and more robots are increasing in the service industry, the inter-robot communication is very critical. This communication can be achieved by ROC and the robots can be monitored remotely or locally via RMS. The RaaS platform will comply with all the standards of SOA (Service Oriented Architecture) like the development platform and execution unit, thereby creating a flexible and more development-friendly process.

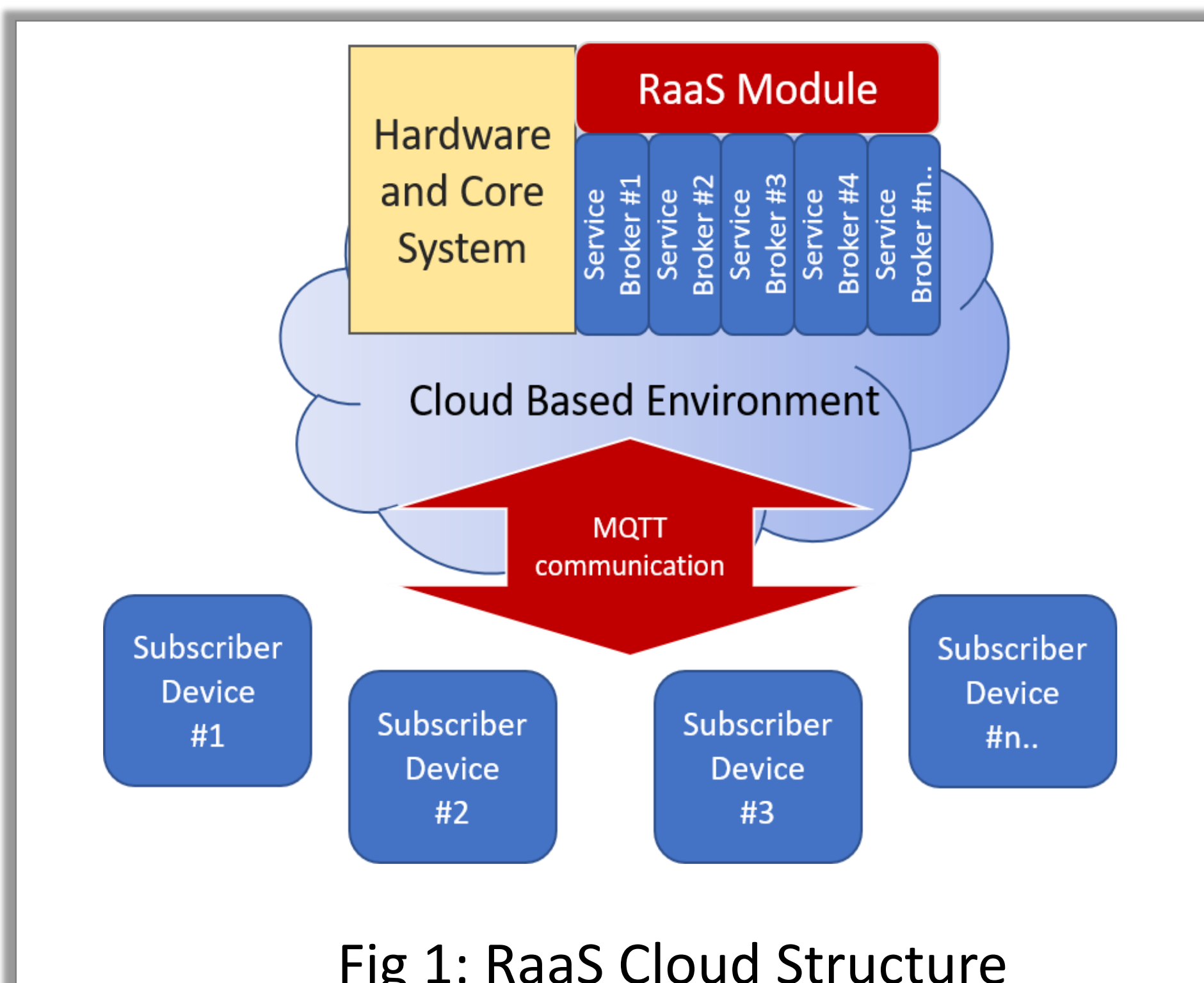


Fig 1: RaaS Cloud Structure

## RaaS Cloud Structure

The Figure 1 explain the overall architecture of the proposed platform. The Service side will do most of the processing and the client side is responsible to record sensory data. The Hybrid solution to this is by processing lightweight task on client side. All the communication takes place using MQTT protocol. Since is MQTT is very lightweight on network, its best suited for this solution.

## Device Interface Architecture

In order to comply with the RaaS architecture a streamline device architecture is needed as shown in Fig 2 which is light weight on processing and also integrates the Robot Operation Centre unit which is essential to RaaS platform, since it is responsible to regulate all the communication with the RaaS server. It contain Device Drivers and Configuration files to manage sensors.

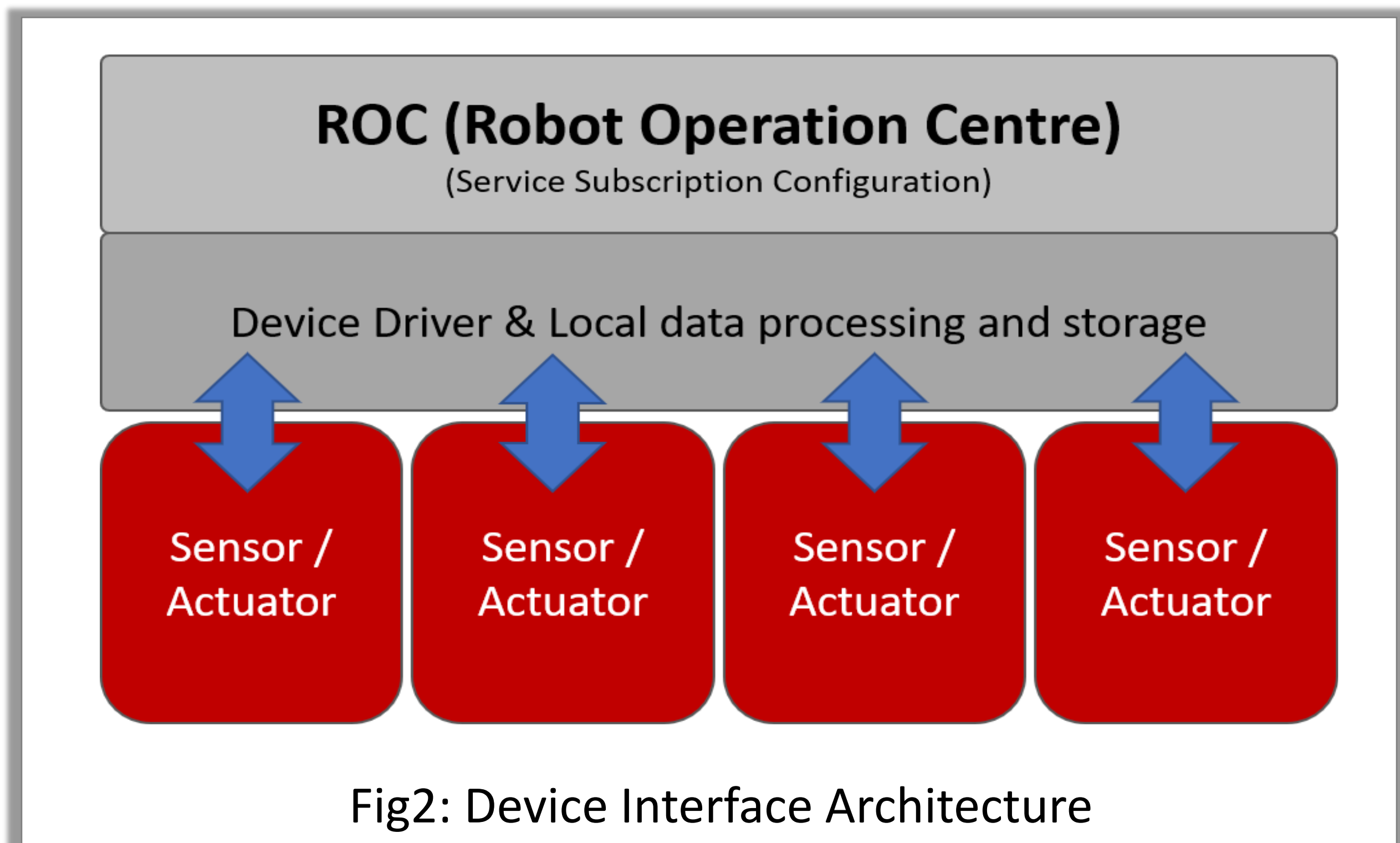


Fig2: Device Interface Architecture

## RaaS Module Architecture

The core building block of the whole RaaS Architecture is the Service directory which are responsible to process all the data fetched by the subscribed devices. As we can see in figure 3 RMS unit is responsible to integrate and aggregate the processed data, so that user can easily monitor and manipulate the devices from remote location.

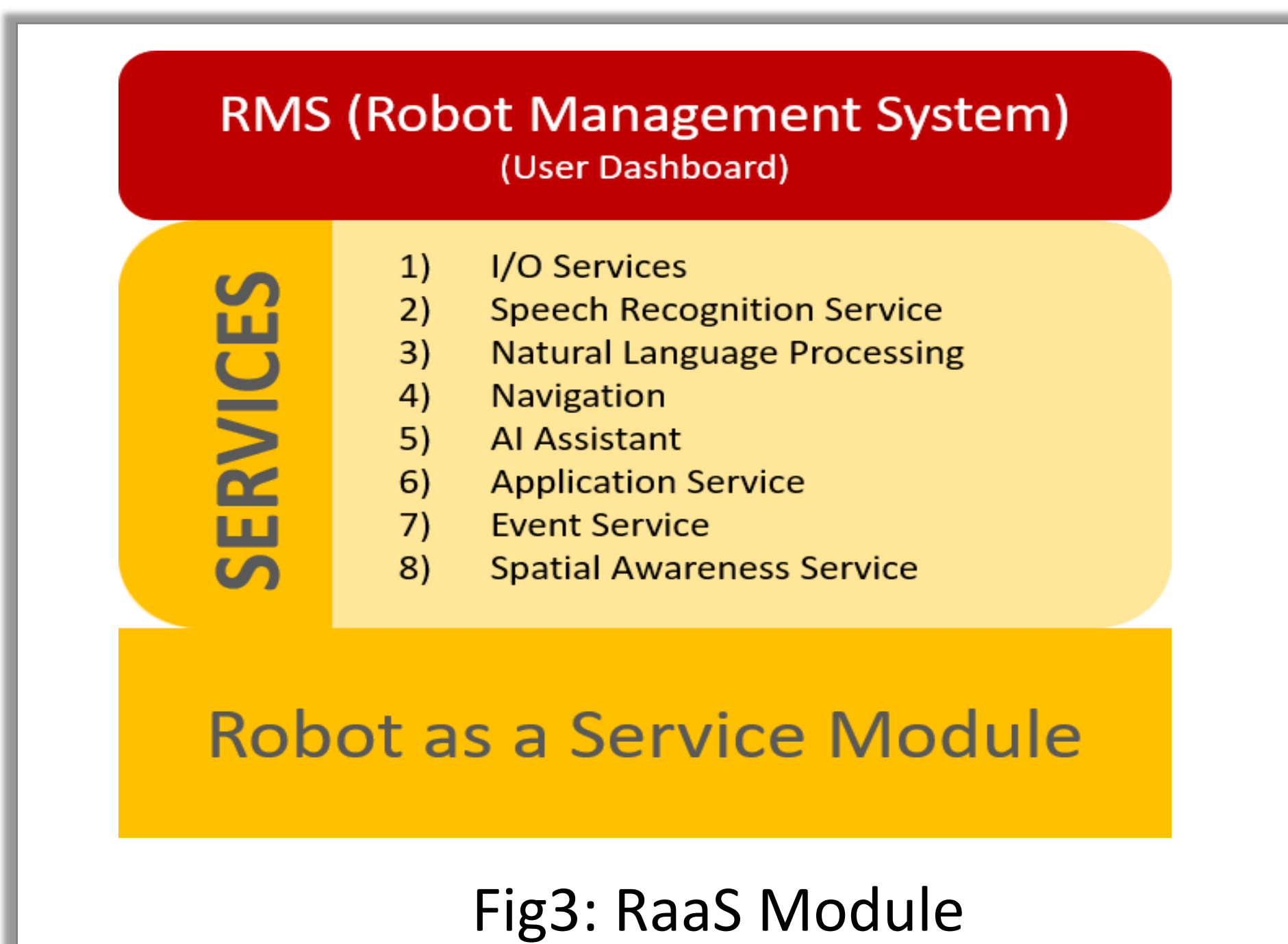


Fig3: RaaS Module

## References

- 1) Costa, Luis F. and Goncalves, Luiz M. G. Universidade Federal do Rio Grande do Norte DCA-CT-UFRN, Campus Universitario, Lagoa Nova, CEP 59.078-970, Natal, RN. luis.feliphe.pb@gmail.com, [Imarcos@natalnet.br](mailto:Imarcos@natalnet.br)
- 2) Yuehai Wang, Jianyi Yang, Robot as a Service in Information Science & Electronic Engineering Education, IEEE 2017