Robot basic architecture based on SSES Blueprint

Architecture document

V0.1 01/28/2022

Table of contents

1. Introduction

Robotics is an important tool for achieving the SDGs. Workers will be able to focus on decent work and new innovations by improvement of labor productivity using robot. As a result, we can move toward new economic growth. However, there are industries where it is difficult to apply current robotics. For example, agriculture, restaurant, food factory, etc.. The biggest challenge current robotics faces in these industries is how to control elastic and non-uniform object under variable circumstance.

The SIP SSES project in Japan tries to solve the challenge through an approach of enhanced cognitive capabilities, new mechanical mechanisms, and enhanced maintenance technologies. They also aim to build CPS by integrating many elements, including AI, IoT, sensor platforms and mechanics. To apply robotics to any industry easily, this blueprint develops and provides open software stack based on the achievements of SSES.

*SIP: The Cross-ministerial Strategic Innovation Promotion Program
SIP is a national program led by the Council for Science, Technology and Innovation （CSTI） of the Japanese Government with interdisciplinary management to realize scientific and technological innovation in our country.
About SIP | SIP | JST

*SSES: Sensor-rich Soft End effector System
立命館大学 SSES Platform - Ritsumeikan Univ. (sip-sses.net)

2. Use cases

Source: SIP SSES project

[公開情報 - Ritsumeikan Univ. SSES platform (sip-sses.net)](sip-sses.net)

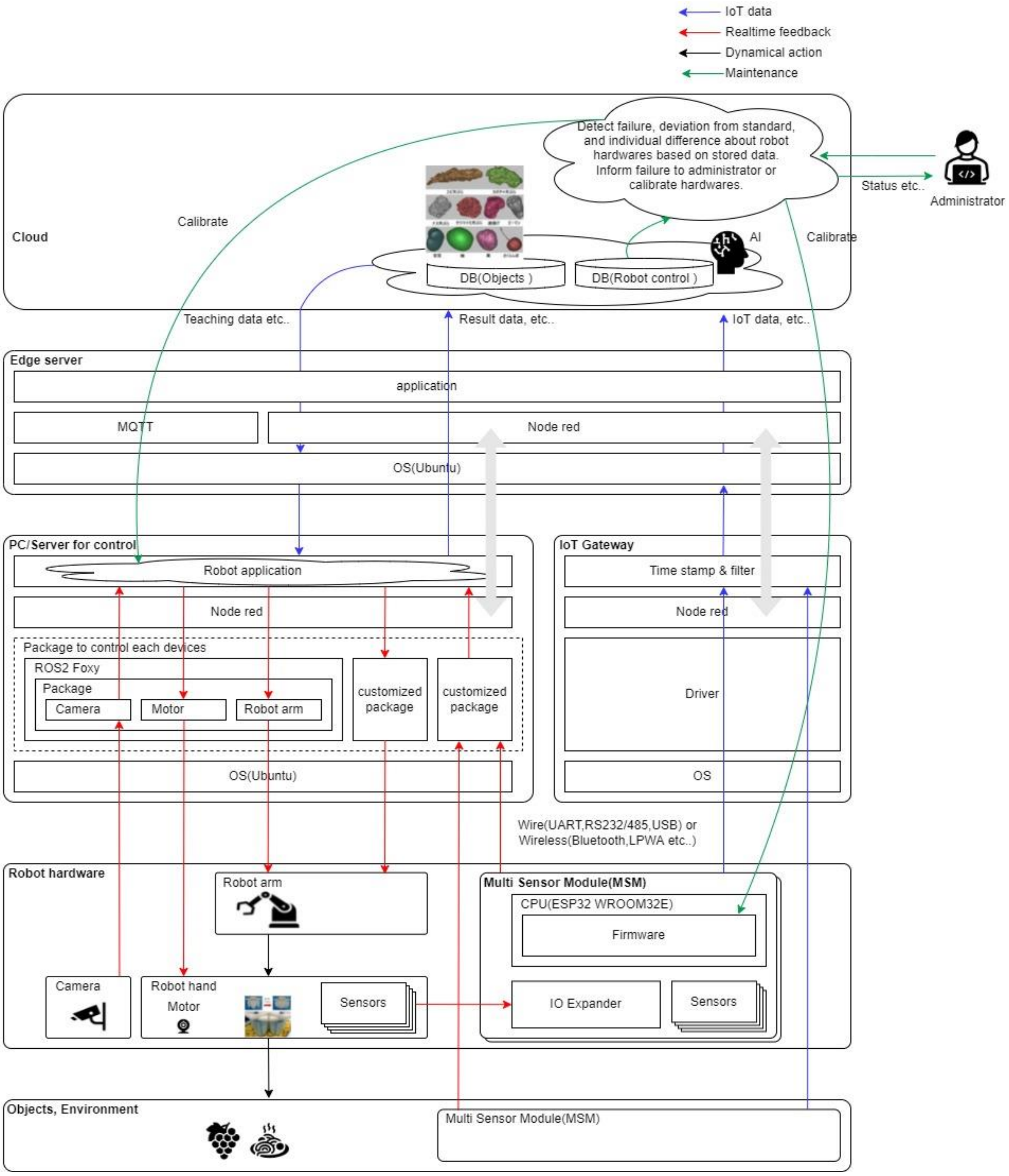
I. Food presentation

II. Put dishes into the dishwasher

III. Automatic collection of tableware

3. Overall Architecture

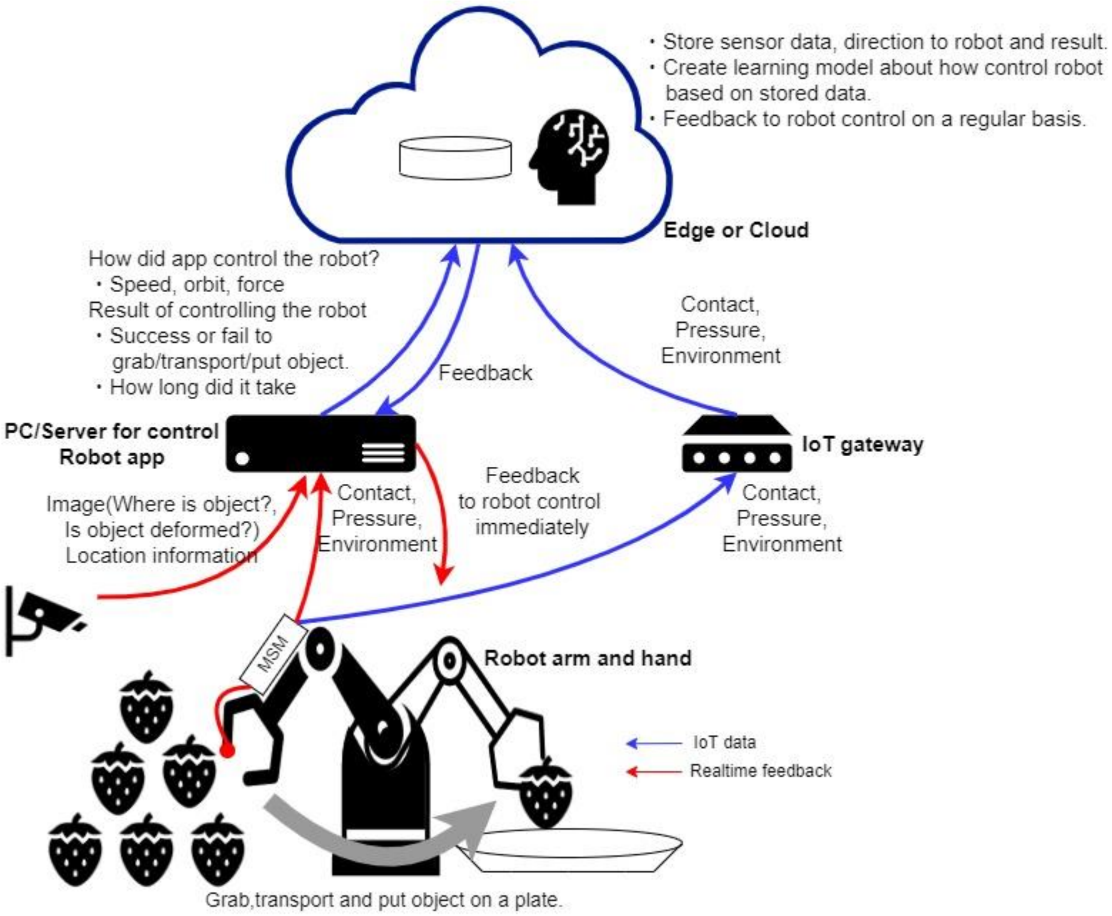
The following figure indicates overall architecture of this blueprint.



- Cloud, Edge Cloud
  - Create data base of IoT data and operation result data
  - Create teaching data for AI based on the data base, and feedback to PC/Server for robot control.
  - Detect failure, deviation from standard, and individual difference about robot hardware based on stored data.
    E.g.
    ・Measure the value of the pressure sensor under no load, and judge whether sensor is broken or not, or there is deviation from standard or not.
    ・Calculate individual values such as the number of revolutions of the robot motor based on time the motor was driven and distance robot moved.
- PC/Server for robot control
  - Recognize objects based on information such as a camera
  - Control arms and hands based on real-time feedback from sensors
  - Optimize grip strength, orbit and speed, etc… based on teaching data from cloud/edge cloud.
- IoT Gateway
  - Shape IoT data from MSM so it can be organized in a cloud database
    E.g. Time stamp, Add units, Scale conversion etc..
- Robot arm
- Hand
- Multi Sensor Module
  - Control various sensors
  - Send information from sensors to PC/Server for control and IoT Gateway via wire or wireless communication.

This architecture is characterized by two data flows "real-time feedback flow" and "IoT data flow". The microcomputer simultaneously distributes the data measured by the sensor in physical space to two systems, the real-time feedback flow "red arrow" and the IoT data flow "blue arrow". The route of the real-time feedback is directly connected to the PC for robot control and used as sensing data for robot control. On the other hand, data flow sends data to the edge server cloud via the gateway to analyze data in the cloud (cyberspace) and provide feedback to robot control on a regular basis. In addition, real-time feedback can be controlled using only the signal strength (gradation value) of the sensor signal, but data analysis on the IoT side requires not only the signal strength of the measured value but also a meaning that matches the physical quantity chemical quantity. Therefore, MSM transmits raw data for low latency. On the other hand, the gateway assigns units to measured values and performs scale conversion according to the units. The following is examples of how to use these two data flows.

E.g.　Food presentation



・Real-time feedback flow

Sensors measure contact, pressure, and other information. Camara provides images for object detection, deformation. Robot app on PC/Server immediately feeds this information back to the robot control.

・IoT data flow

Edge or cloud store data measured by sensors via IoT gateway. Addition to that, edge or cloud store data about how app control the robot and result.

E.g. Speed, orbit, force

Success or fail to grab/transport/put object,

How long did it take.

Edge or cloud create a learning model to control the robot more optimally and feed it back to robot control less frequently than real-time feedback.

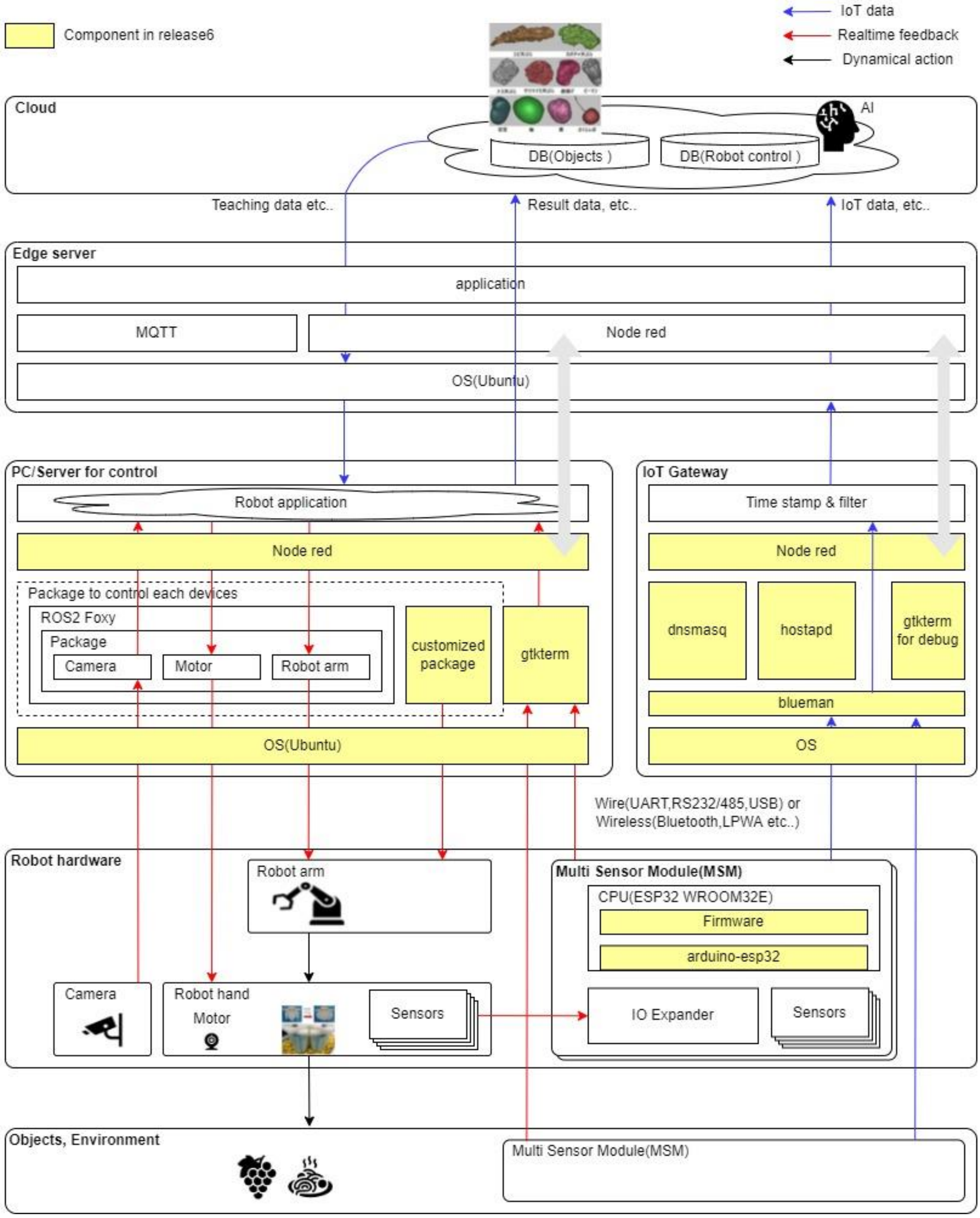
4. Platform Architecture for release6

Component	Hardware name	
PC/Server for control		Will update before the test document is fixed.
IoT gateway	Raspberry pi 4model B	
MSM	Pre Multi Sensor Module	https://sip-sses.net/quick-reference/
Robot arm		Will update before the test document is fixed.

5. Software Platform Architecture for release6

Component	Software name	Version
PC/Server for control	Ubuntu	Will update before the test document is fixed.
	Node red	Will update before the test document is fixed.
	Robot arm package	Will update before the test document is fixed.
	gtkterm	Will update before the test document is fixed.
IoT gateway	Raspbian	Will update before the test document is fixed.
	Node red	Will update before the test document is fixed.
	gtkterm	Will update before the test document is fixed.
	hostapd	Will update before the test document is fixed.
	dnsmasq	Will update before the test document is fixed.
	blueman	Will update before the test document is fixed.
MSM	arduino-esp32	Will update before the test document is fixed.
	preMSM firmware	v1.13

The following figure indicates platform and software architecture in release6.



6. API

   NA

7. Hardware and Software Management

   Software

Component	Software name	repository
PC/Server for control	Ubuntu	Will update before the test document is fixed.
	Node red	Will update before the test document is fixed.
	Robot arm package	Akraino repository Will update when we commit.
	gtkterm	Will update before the test document is fixed.
IoT gateway	Raspbian	Raspberry Pi OS – Raspberry Pi
	Node red	Will update before the test document is fixed.
	gtkterm	Will update before the test document is fixed.
	hostapd	Will update before the test document is fixed.
	dnsmasq	Will update before the test document is fixed.
	blueman	Will update before the test document is fixed.
MSM	arduino-esp32	https://github.com/espressif/arduino-esp32
	preMSM firmware	Akraino repository Will update before the test document is fixed.

8. Licensing

Software

Component	Software name	License name
PC/Server for control	Ubuntu	GPL
	Node red	Apache License2.0
	Robot arm package	Apache License2.0
	gtkterm	GPL 3.0
IoT gateway	Raspbian	BSD 3Clause,MIT,GPL
	Node red	Apache License2.0
	gtkterm	GPL 3.0
	hostapd	GPL 2.0
	dnsmasq	GPL 2.0 or 3.0
	blueman	GPL 3.0
MSM	arduino-esp32	LGPL 2.1
	preMSM firmware	Apache License2.0

Revision history

Version	Date	Editor	Contents
0.1	01/28/2022	Fukano	Draft version