Sustainable Computing: Informatics and Systems 28 (2020) 100285

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

journal homepage: www.elsevier.com/locate/suscom

A Creative IoT agriculture platform for cloud fog computing

Tse-Chuan Hsu [a,b,*], Hongji Yang [c], Yeh-Ching Chung [d], Ching-Hsien Hsu [e,f]

[a] Centre for Creative Computing, Bath Spa University, Bath, England, United Kingdom
[b] Hsuan Chuang University, Hsinchu, Taiwan
[c] Department of Informatics, Leicester University, Leicester, England, United Kingdom
[d] School of Science and Engineering, Chinese University of Hong Kong in Shenzhen, Shenzhen, Guangdong 518172, China
[e] Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan
[f] Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan

ARTICLE INFO

Article history:
Received 16 August 2018
Received in revised form
22 September 2018
Accepted 20 October 2018

Keywords:
Agriculture
Creative platform
Fog computing
Communication model
Internet of things

ABSTRACT

The innovative service process is a process that uses newly developed technologies to improve the current service models. The study proposes a creative service process based on the cloud computing platform of the Internet of Things and it can be used to improve the integration of the current cloud-to-physical networking and to improve the computing speed of the Internet of Things. In the past, most of the computing technologies focused on high-speed computing in cloud computing or remote operations of a single object. If a service requires cloud or fog resources that can make device use of high-speed computing in the cloud and a single point of operation integration on the object side, it will be able to quickly increase in the process of collaboration, the required data will be moved back and forth between Cloud and Fog, speeding up the cloud computing integration schedule. This research uses innovative platform technology to be applied to the cloud agriculture platform. Through cloud integration, it can be applied to large-area data collection and analysis, allowing farmland with limited network information resources to be integrated and automated, including agricultural monitoring automation, pest management image analysis and monitoring, which can be used to solve the predicament of large-area automation construction.

1. Introduction

At present, the development of the monitor system is quite diverse, including traditional CCTV monitor systems, network camera systems, 4 G wireless networks, etc. All images are transmitted back to the management monitoring server through the communication system [1]. After all image retrievals need to be re-imaged, needs manual review of each file, if you want to view multiple monitor images at the same time, it will take a lot of time and network costs. The road surveillance systems in various countries are quite numerous and complicated. When illegal elements ran away after a robbery incident, the police needed to mobilise numerous manpower to carry out video surveillance of all CCTV systems while conducting investigations to find out the suspects. Missing can capture the prime time the criminals have not escaped.

In the traditional IOT environment, it can be found that one-to-one data matching and image capture can help the processing solved the data collection problem [4], including remote camera operations, data storage, remotely reconfiguring, etc. However, if there are more equipment's in the on-site environment, like a traffic intersection, there are more than 8 cameras. Each setting and storage needs to be set 8 times, which is very inconvenient for the operators. Moreover, if you need to read and compare data in an emergency, you will be very anxious. For example, when people are robbed on the roadside. After the someone reported to police offer, they must wait for the CCTV to connect, and check the files one by one to compare cameras, and through the Internet, first step needs pass the image back to the near end, often missed the emergency processing time.

In the current cloud system, a cloud-based storage architecture can help devices upload data and manage data information in a centralised manner. However, if there are many images that need to be uploaded and stored at the same time, network traffic will be heavy and take more times, regardless of bandwidth management [2,3]. As well as server storage management, it is a model with high management costs and inconvenient operation procedures.

---

[*] Corresponding author at: Centre for Creative Computing, Bath Spa University, Bath, England, United Kingdom.

E-mail addresses: davidhsu@hcu.edu.tw (T.-C. Hsu), Hongji.Yang@Leicester.ac.uk (H. Yang), ychung@cs.nthu.edu.tw (Y.-C. Chung), chh@chu.edu.tw (C.-H. Hsu).

https://doi.org/10.1016/j.suscom.2018.10.006
2210-5379/

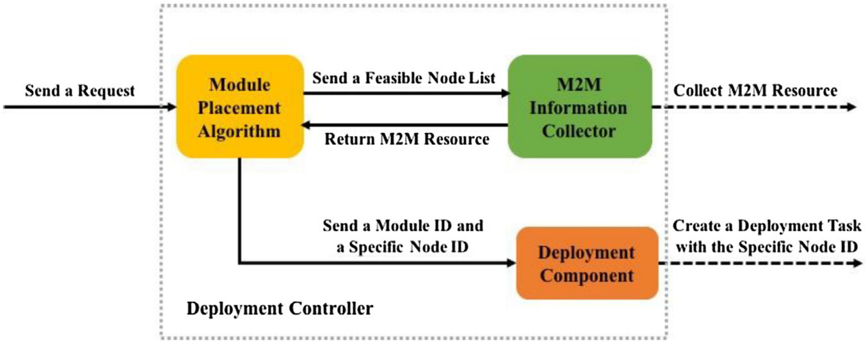


Fig. 1. The module deployment flow in deployment controller.

In this study, a decentralised calculation method was designed to attempt to provide nodes with analysis and judgment capabilities in all nodes, to construct a cloud-side and fog-end-to-cloud mutual service, used IoT perform connect data integration to cloud system [5]. The construction of an inter-layer system for operating the communication mode between the fog-end devices, this architecture can solve the existing IoT real-time data processing problems. In order to solve the above-mentioned problems, data collection is performed in large image files systems, and the comparison of data is completed after completion, which takes for a period of time. This study we discuss implementing discovery and solving challenging problems. It reviews relevant research and development that currently exists in the most advanced technologies that contribute to computer-aided systems. It seeks a new way to convert through a platform for constructing mechanisms. The method of centralised and decentralised processing analysis, mechanism and method corrections, speed up and improve the comparison performance. It can reduce the time required for network transmission, so that in the event of an emergency, system can find the answer you want to know more quickly and effectively. The article proposes an innovative service model, uses the Institute to propose a mechanism platform, establishes creative methods and corresponding creative services, development can use this mechanism to provide different service applications. Through this research mechanism platform, it provides decentralised operations that cannot be provided in the past.

2. Background

In the past, developers can use the PAAS Layer to develop system programs through the cloud platform. The development of program systems can be performed without the need for hardware level on I/O operation and management of the underlying devices. The system will automatically allocate the required hardware, provide software programs to compile and service, and then use the SAAS layer for service applications [5,7]. This model can quickly reach developer system development and service applications for different cloud base layer systems. Since we use of the PAAS layer system development model [16], many companies have also invested in developing tools and providing services. For example, the Heroku system is based on the Salesforce platform and can be used to develop various kinds of web application systems, including web server system operations. For example, many companies in the PaaS layer have provided links including related and non-associative database links, mail server modules, etc. After years of evolution, they have provided developers with a convenient development environment.

The cloud platform system has data analysis capabilities, large storage management space, and a unified management platform capability for immediate dispatch of task events [20]. If the IoT and cloud system platform are connected in series, the IoT device is set to the communication node and designed with a fog architecture [21]. The core architecture layer connects the IoT communication layer with the PaaS layer in the cloud. As a fog network computing node, computes the SaaS layer application service that is assigned to the task and transmits it to the object through the fog [19].

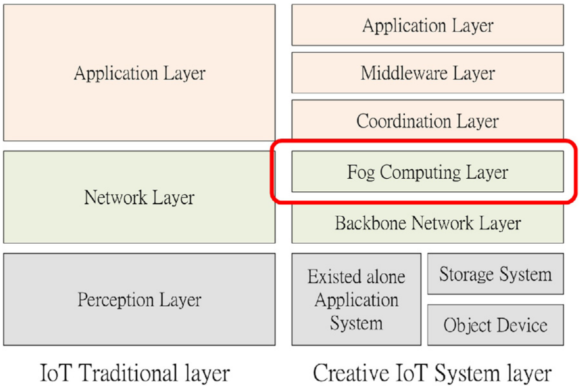


Fig. 2. Traditional IoT three-layer structure with Creative IoT system layer.

In M2M framework design [6,9–12], when user sends a request which needs a specific module deployed near the user, deployment controller which is at the top of M2M framework will receive the request. And then, the Module Placement Algorithm (MPA) will collect the M2M resource of the related nodes through M2M Information Collector according to the request and make module placement decision in order to maximise the reduction of M2M network traffic and load of data centers [13]. After making the module placement decision, deployment component will create a deployment task with a specific node ID and perform the deployment task with the specific node's deployment component. [8] In Fig. 1, we show the simple steps of module deployment flow in deployment controller.

3. The design for Creative IoT platfrom

In the cloud system, a standard architecture has been proposed and developed, including the three-tier structure of IaaS, PaaS, and SaaS. When the Internet of Things is integrated with cloud technologies, the required environment is not limited to the traditional three-tier structure. About network communication structure, it is necessary to strengthen the communication and operation processing characteristics and establish an atomisation architecture. Fig. 2. shows that in the cloud application system, the traditional classification [14,15] is performed through three layers, and the hardware is supplied by the device. PAAS is a software development tool that assists the developer in system development and finally provides application services into SaaS. If we want the Internet of Things and the Internet to communicate in addition to the traditional communication, let the devices be connected to each other and achieve a decentralised computing model, it needs to consider the design of the communication logic structure.

It needs for a new method in the physical networking environment. Therefore, a method has been proposed in the research to improve the internet of things device problems. In the IoT service, in addition to the basic hardware layer, in the design phase, this study established a Creative IoT Service Platform mechanism to design an innovative service architecture platform model, Fig. 2. design and Improved the existing PAAS layer architecture in the cloud and incorporated fog computing into the PAAS. Analyse and calculate through layers, design through fog computing mechanism, solve the Internet of Things calculation, the real-time information generated by the equipment uses a decentralised calculation method to accelerate and reduce the incident response time, and at the same time integrate the communication mechanism with the PAAS, in addition to the traditional data transmission Apart from technology, this method can be more improved and breakthrough than the traditional three-tier application of Internet of Things [16–18].

4. Decentralised internet of things data processing model

In cloud and fog integration, the traditional Internet of Things (IoT) device sensing layer, including the operating system and I/O service, is retained on the device side for end-point object operation and application. In the communication layer, it addition to the previous network layer for data communication, the upper layer assists in the packaging of application service results for network transmission. In the research, a concept of a smart and innovative service was proposed to put fog calculations into the IoT communication layer. Fog Computing is mainly used to manage data from sensors and edge devices. It centralises data, processing, and applications in devices at the edge of the network rather than all in the cloud data center. We designed a decentralised computing method through fog computing technology to distribute computing, communication, control, and storage resources and services to users or devices and systems close to users. The integration targets the lack of IoT device and cloud computing, and the proposed solution can be integrated into the IoT fog computing to integrate the three-tier architecture of the Internet of Things and add a layer of services to the network connection layer, expanding the traditional master-slave mode. The network computing model of the architectural communication model calculation is more widely applied to various services (Figs. 3 and 4).

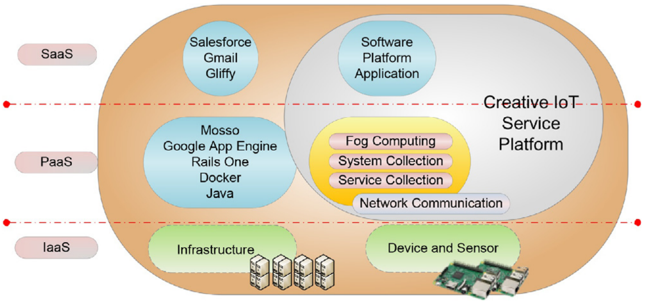


Fig. 3. Creative IOT Service Platform.

4.1. Hardware design with creative platform

In order to provide all devices in series, this study designed an innovative and intelligent IOT cloud service platform. The service platform must be combined with software and hardware to effectively provide data collection and software command service event management. In the planning of design hardware equipment; This study we design of the two major areas of service, the first one for the centralised management of the interim hardware design. Since the hardware architecture is shown in Fig. 5, a set of mediation systems was designed in the study. This mediation system can be used to communicate with IoT terminal devices to exchange information in a small area, such as a connection network environment that can meet the requirements of the LAN. Under the regional connection environment, the device can communicate with the gateway through a wireless network or a physical network. The gateway will serve as data synchronisation, event dispatch, and program software data collection. When the same service is used in the WAN environment, it is automatically connected to the device through the wide-area network through the cloud mediation platform. The second is single-point component design. The component design architecture will establish the IoT communication module. The main task is to design the embedded hardware system so that the IoT device can communicate with the gateway through the communication module. Network communication modules needs to have a two-way interactive operation model. When the device is in the Internet of Things environment, it will automatically find the

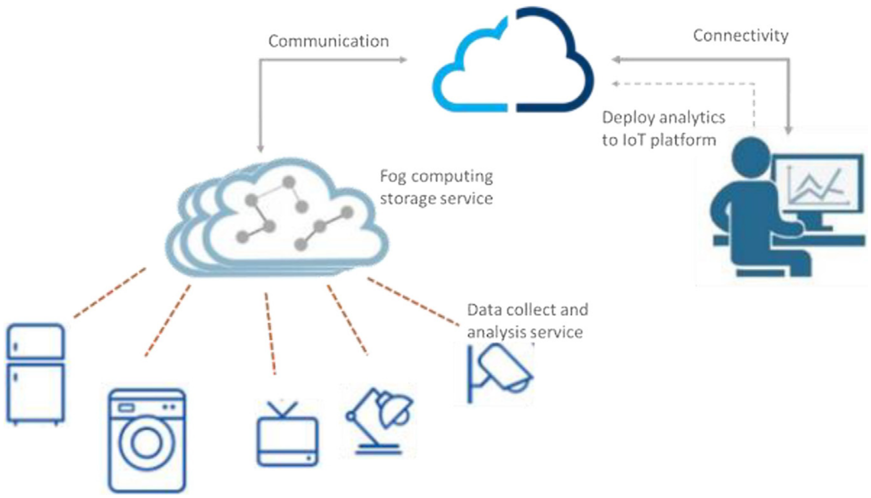


Fig. 4. Fog Service with IoT platform.

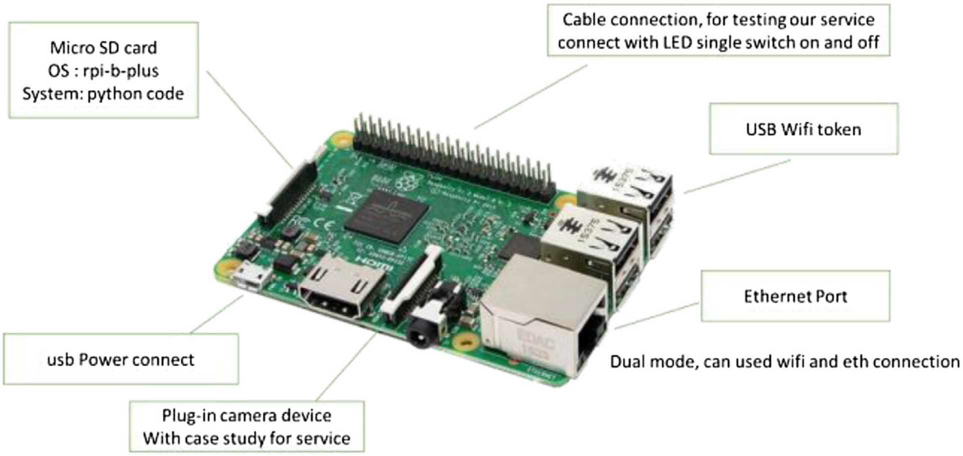


Fig. 5. Creative IOT Service Platform.

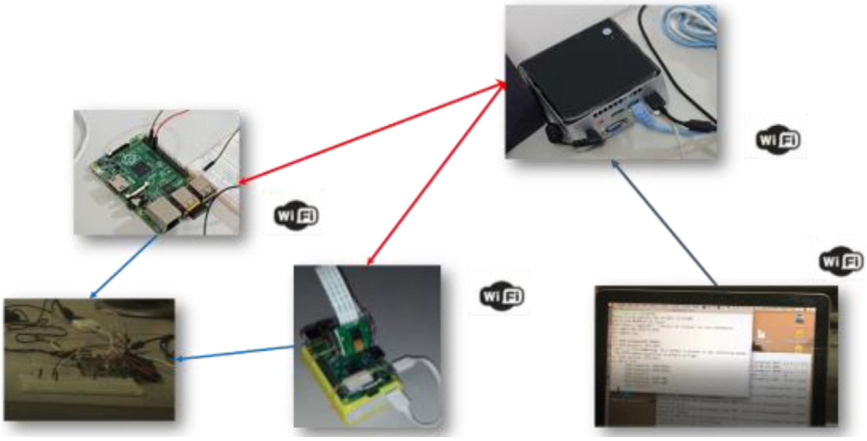


Fig. 6. IOT Box and Device Connection Architecture.

gateway device, and the gateway can provide real-time monitoring and management communication services.

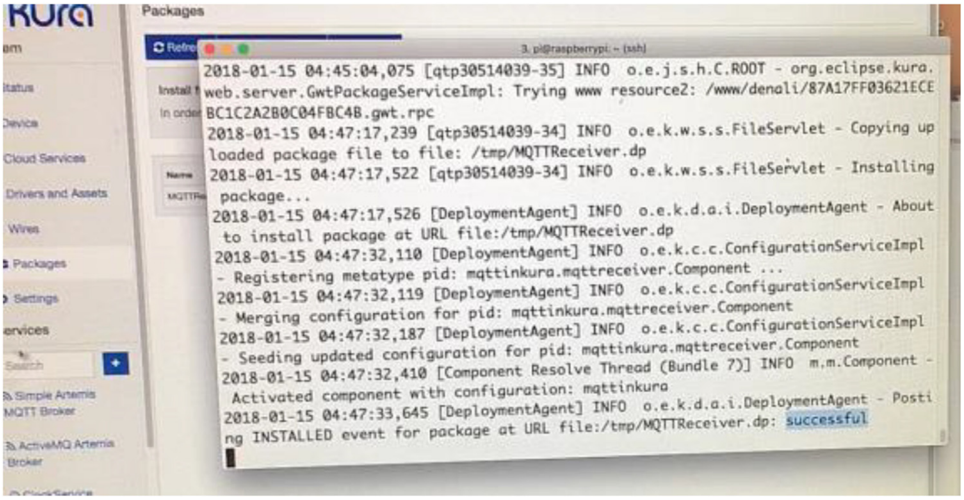
In hardware devices, Raspberry Pi is used for architectural design. The reason is that it is an independent operating system platform, which facilitates the development of experimental simulation development systems, and also uses the underlying operating system and design of the Linux operating system. When the data is collected, and the parallel data storage or synchronisation mechanism can be used to write the analysis data into the database. In the networked device environment, I/O access and data analysis can be performed independently.

In the study, reference is made to the OSGi basic standard and an OM2M model is established. The research and design is shown in Fig. 6, In order to provide effective communication between the equipment and the gateway, both the gateway and the IoT device have developed and established the same communication agent. System can be waiting and get communication on the Internet. The hardware device on the object side automatically starts the client agent to automatically find the gateway service after the device is start on. After the gateway device is start on, it automatically completes the establishment of the communication mechanism and waits for the gateway to send the task to the device, and the task is dispatched. The software program system including the use operation can be dispatched, and the triggered work dispatches the task event. When the event is required to be executed, raspberry pi will send device of the Internet of Things terminal device will execute the event task according to the gateway command request. Fig. 7, shows the hardware side waiting for receiving event program and operation screen. When the gateway receives a task assignment, it dispatches the task. Fig. 8, represents the successful delivery of the gateway task. Fig. 9, shows the object-side device of the Internet of Things, receives the instruction and executes the operation, and sends the result back to the IOT creative platform.

4.2. Software design for IoT platform

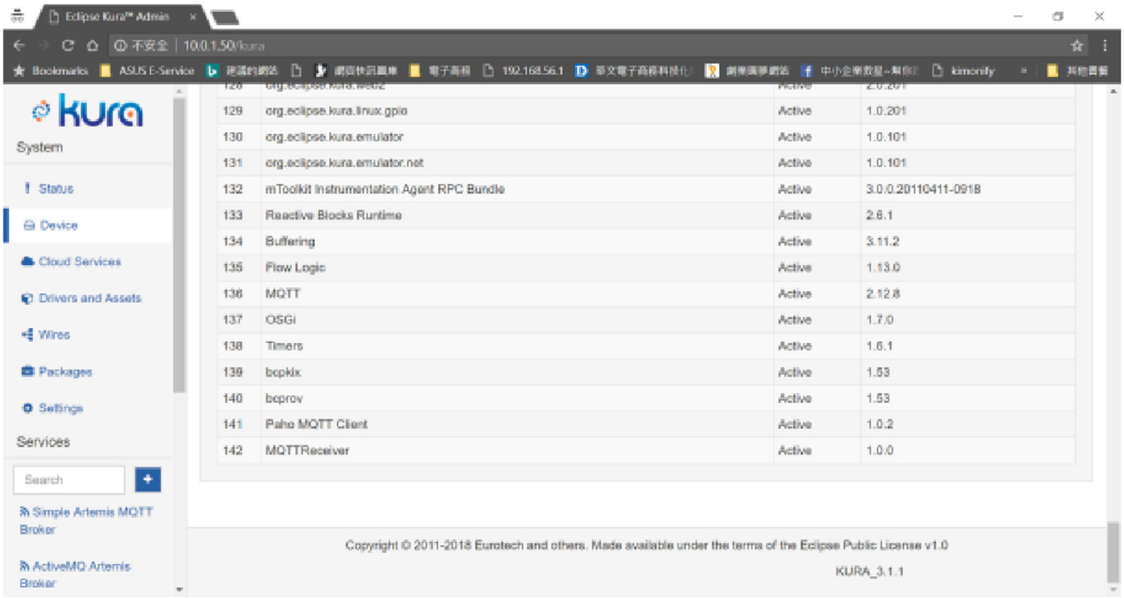
For software system architecture diagram, we design and implementation of middleware, an intermediary software that conforms to the environment of the Internet of Things. Through this intermediary software, the system service status monitoring and service configuration integration ensure the provision of user-required corresponding services. For example, when the user needs to quickly assist in computing different camera image information, the software system performs task broadcasting through the core platform, automatic select a regional camera for analysis and compare data information. The software system will assign tasks to agents of all devices. System agent performs work for the assigned tasks and returns task messages back to the platform. Fig. 10, shows that in the service port, the platform core system and the device on the object side perform proxy information exchange and exchange services, and use the OSGi M2M standard specification to enable different devices to operate through agents.

In order to construct the problem of achieving the above explanation, the scope of research in this study is to propose an



```
2018-01-15 04:45:04,075 [qtp30514039-35] INFO  o.e.j.s.h.C.ROOT - org.eclipse.kura.
web.server.GwtPackageServiceImpl: Trying www resource2: /www/denali/87A17FF03621ECE
BC1C2A2B0C04FBC4B.gwt.rpc
2018-01-15 04:47:17,239 [qtp30514039-34] INFO  o.e.k.w.s.s.FileServlet - Copying up
loaded package file to file: /tmp/MQTTReceiver.dp
2018-01-15 04:47:17,522 [qtp30514039-34] INFO  o.e.k.w.s.s.FileServlet - Installing
 package...
2018-01-15 04:47:17,526 [DeploymentAgent] INFO  o.e.k.d.a.i.DeploymentAgent - About
 to install package at URL file:/tmp/MQTTReceiver.dp
2018-01-15 04:47:32,110 [DeploymentAgent] INFO  o.e.k.c.c.ConfigurationServiceImpl
- Registering metatype pid: mqttinkura.mqttreceiver.Component ...
2018-01-15 04:47:32,119 [DeploymentAgent] INFO  o.e.k.c.c.ConfigurationServiceImpl
- Merging configuration for pid: mqttinkura.mqttreceiver.Component
2018-01-15 04:47:32,187 [DeploymentAgent] INFO  o.e.k.c.c.ConfigurationServiceImpl
- Seeding updated configuration for pid: mqttinkura.mqttreceiver.Component
2018-01-15 04:47:32,410 [Component Resolve Thread (Bundle 7)] INFO  m.m.Component -
 Activated component with configuration: mqttinkura
2018-01-15 04:47:33,645 [DeploymentAgent] INFO  o.e.k.d.a.i.DeploymentAgent - Posti
ng INSTALLED event for package at URL file:/tmp/MQTTReceiver.dp: successful
```

Fig. 7. MQTT receiver listening.



129	org.eclipse.kura.linux.gpio	Active	1.0.201
130	org.eclipse.kura.emulator	Active	1.0.101
131	org.eclipse.kura.emulator.net	Active	1.0.101
132	mToolkit Instrumentation Agent RPC Bundle	Active	3.0.0.20110411-0918
133	Reactive Blocks Runtime	Active	2.6.1
134	Buffering	Active	3.11.2
135	Flow Logic	Active	1.13.0
136	MQTT	Active	2.12.8
137	OSGi	Active	1.7.0
138	Timers	Active	1.6.1
139	bcpkix	Active	1.53
140	bcprov	Active	1.53
141	Paho MQTT Client	Active	1.0.2
142	MQTTReceiver	Active	1.0.0

Copyright © 2011-2018 Eurotech and others. Made available under the terms of the Eclipse Public License v1.0

KURA_3.1.1

Fig. 8. MQTT receiver service log.

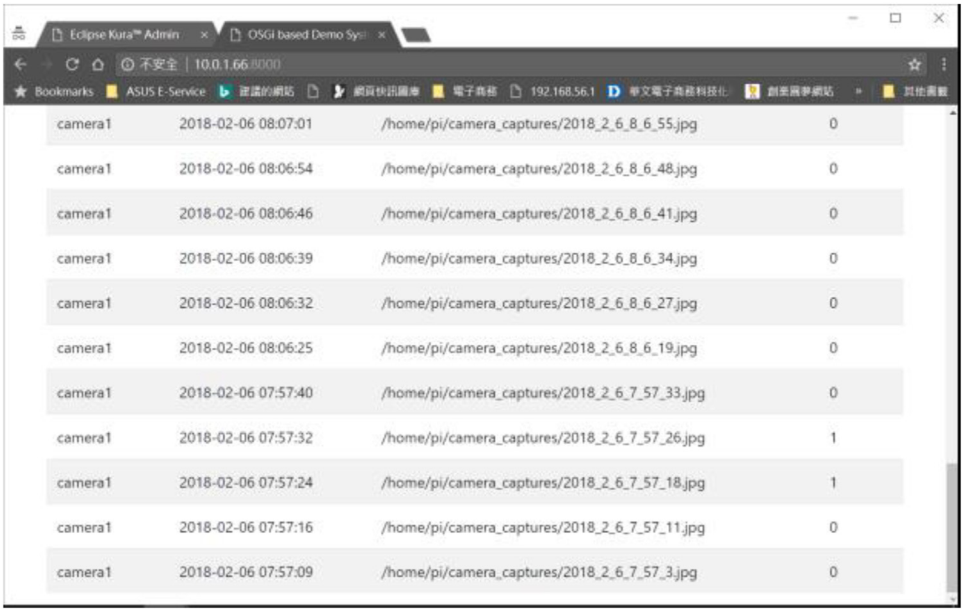
innovative IOT service platform that can provide communication between object devices. An innovative service platform we propose is to design a set of innovative intelligence logic after devices and devices communicate with each other so that devices can know which roles assigned to and help the entire IoT environment. In this study, try to find out how to create a platform with a changeable innovative service. After different devices are connected with the platform, you can understand the tasks that can be performed by themselves and assist the operators in establishing an innovative service application model.

5. Experiment instructions

In the research, a Creative IoT Service Platform was constructed through experiments. Based on the OSGi standard protocol, the OM2M feature of the Internet of Things (IoT) was used in the agreement to design a set of IoT devices that can be used for future innovative application services.

The main design principle of Creative IoT Service Platform is to establish the Internet of Things communication service agent engine and construct the agent management mechanism. The following explains that the system design architecture includes hardware device communication, network data transmission and application service system installation and operation. The three key point of technologies are designed and the data exchange format XML it can used to transmit the service operation results to various information service platforms for subsequent application. For the agent-side system design of the device, in the research, the agent is installed on the terminal device and communicates with the platform via the terminal device. Fig.11, When the service platform has specific software service application requirements, the data can be transmitted through the network message exchange and publish install and respond to the installation status and service conditions.

Fig.12, Taking the simulation case of this research system as an example, the Raspberry Pi is used to simulate the IoT device and perform hardware device operation services. In the device operation, the raspberry pi camera is used to simulate the monitor behavior mode, let the device simulates remote dispatch management operations. In the end point device, the MQTT client is installed first on the device endpoint. For example, this agent sys-



camera1	2018-02-06 08:07:01	/home/pi/camera_captures/2018_2_6_8_6_55.jpg	0
camera1	2018-02-06 08:06:54	/home/pi/camera_captures/2018_2_6_8_6_48.jpg	0
camera1	2018-02-06 08:06:46	/home/pi/camera_captures/2018_2_6_8_6_41.jpg	0
camera1	2018-02-06 08:06:39	/home/pi/camera_captures/2018_2_6_8_6_34.jpg	0
camera1	2018-02-06 08:06:32	/home/pi/camera_captures/2018_2_6_8_6_27.jpg	0
camera1	2018-02-06 08:06:25	/home/pi/camera_captures/2018_2_6_8_6_19.jpg	0
camera1	2018-02-06 07:57:40	/home/pi/camera_captures/2018_2_6_7_57_33.jpg	0
camera1	2018-02-06 07:57:32	/home/pi/camera_captures/2018_2_6_7_57_26.jpg	1
camera1	2018-02-06 07:57:24	/home/pi/camera_captures/2018_2_6_7_57_18.jpg	1
camera1	2018-02-06 07:57:16	/home/pi/camera_captures/2018_2_6_7_57_11.jpg	0
camera1	2018-02-06 07:57:09	/home/pi/camera_captures/2018_2_6_7_57_3.jpg	0

Fig. 9. Return analysis results, image files and status values.

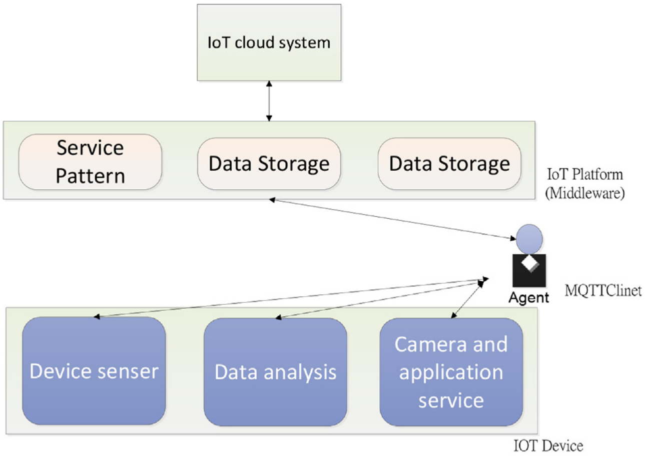


Fig. 10. IoT Platform Middleware Architecture.

tem will wake up automatically each time the device is started and stays at the back end of the system. It is connected to the Creative IoT Service Platform. network, waiting for notification to dispatch the task.

Experiment instruction using software development technology, an OpenCV image analysis application system was imported, the first delivery was performed through the IoT platform. Fig.13, Using the synchronous distribution model, the application system was installed to provide connection materials on the device endpoint. When setting up a script, we need to send a camera to take pictures and analysis services, upload the communication service code on the platform, and send the code to the IoT device for installation via the distribution mechanism, such as camera.py. The main program software services, when task delivery is performed through the IoT service platform, the system will assist in launching the Raspberry pi camera for script design. In this experimental example program, the camera will be shot each time the camera is opened, and the image will be shot and the OpenCV analysis will be started after the image is taken. The information is sent back to the message and photos. If there is no one, 0 message is returned, and the record is completed in the system.

5.1. Experimental results

Using the IoT Platform to simulate the corresponding comparison, we discussed the verification time for a single device operation in the traditional IoT mode, and the data storage and analysis technology application service mode of the object networking device in Table 1.

First, in the device port, study and set several basic conditions, such as the above table, A1 represents each IoT device after the camera started, the resulting data type, A2 quantitatively set the size of each data files, A3 for network Under the environment, set the upload throughput 50 mb/per min, A4 is the time required to install the service application system, A5 sets the task assignment behavior to the time required by the device, and A6 provides the total time required for data collection. Area B represents the remote server's operation processing conditions. Under the traditional mode environment, all data are uploaded and stored for image analysis. B1 is set to download time through the network and is set to download throughput 100 mb/per min. B2 is the server processes the analysis time for each image files.

When using traditional data storage methods for operation, data manipulation and storage are performed on a single device and image events are processed for 1-4 hours. Table 2 Can get the results obtained are as follows:

IoT Creative Platform was used to perform calculations. Through the same environmental condition parameter status, the agent service connection was first improved, and the computing environment platform was used to perform cloud data exchange. The experimental results are as follows in Table 3.

The experimental results show that using the IoT platform, the longer the detection time is, the lower the analysis time cost will be. The reason of the fog performs calculation analysis at the near end, take recording times longer compared with the traditional model. When recording times longer and analysis time cost will change. Smaller, compared with 4 hours, the average growth rate decreased from 32 minutes to 24 minutes compared with the traditional model. Since the number of network backhaul files was reduced, and the network cost time was reduced. After four hours of data analysis, the required time cost was the traditional model average. It takes approximately 69 minutes and can be reduced by about 39 minutes using the IoT Creative Platform. It can be seen

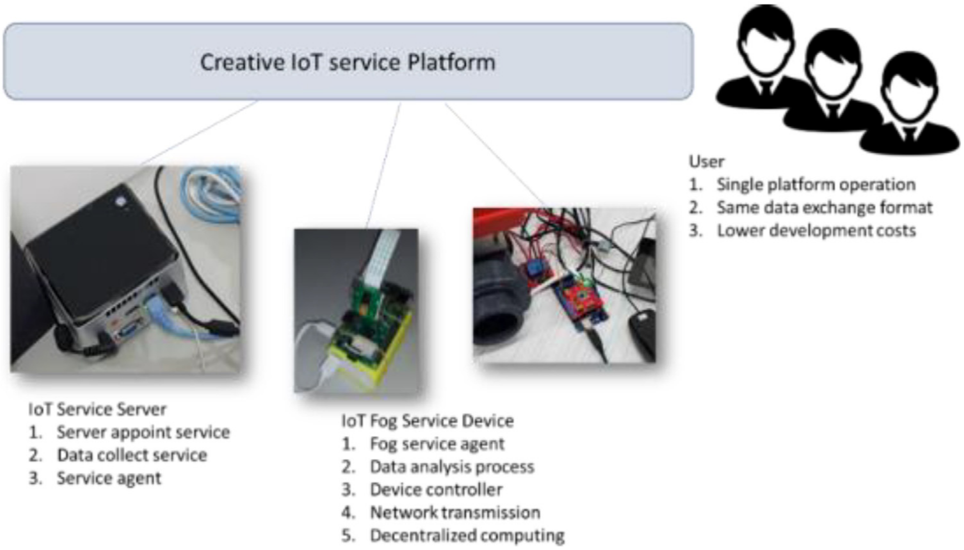


Fig. 11. Simulation platform design system architecture.

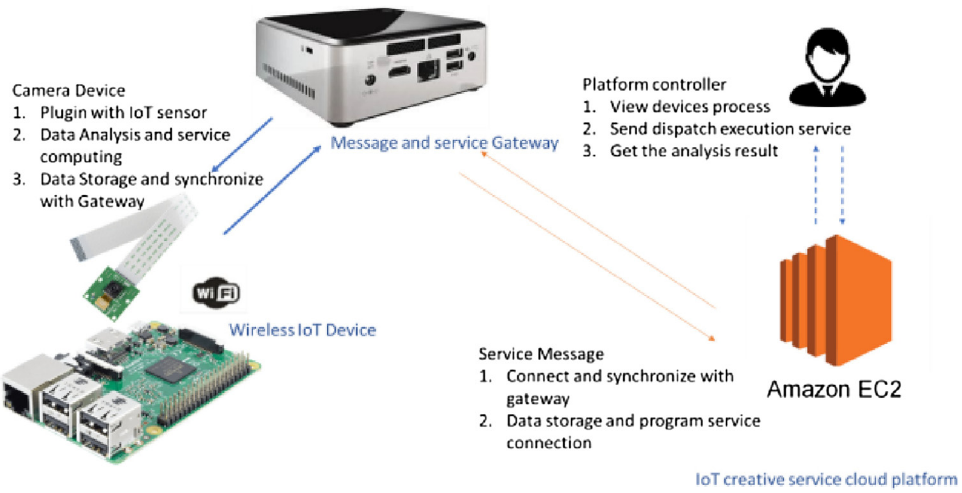


Fig. 12. Raspberry Pi Camera Surveillance Camera.

that under the state of fog calculation, the longer the continuous use analysis time is, the more significant the improvement.

6. Experiment innovation agriculture platform

In the research, through real-time image monitoring management design, combined with temperature and humidity sensing sensors, real-time data management, including soil pH, sun exposure time, disease information collection, further integration of innovative IoT smart automated agricultural disease management services construct pest and disease control as the core innovative service application system in Fig. 14).

Fig. 15, shows a set of instant pest detection systems based on image alignment analysis. When the disease occurs, the plant growth status characteristics are written into the database and compared with the on-site status. Through the innovative service platform, all fog devices are automatically controlled to perform image comparison results. When the collected data cannot be judged, it is automatically transmitted back to the IoT creative service platform through the middleware for tracking. If the information obtained after the growth is determined, the feature will affect the plant growth disease, and then stored in the disease database. At the same time, the feature image conditions to be compared are sent to the fog for subsequent image data comparison operations.

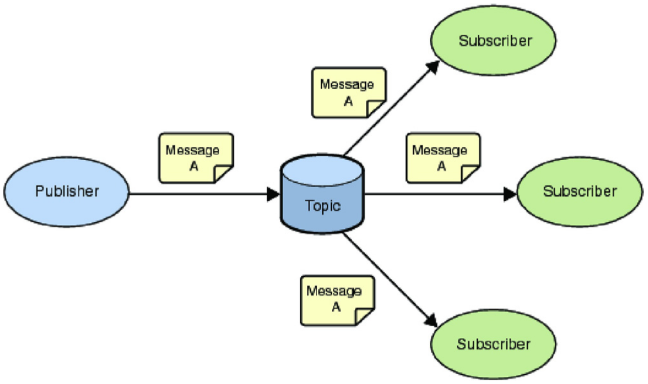


Fig. 13. MQTT client resident agent system.

In the research system, the experiment uses the sensing device to adjust the temperature and humidity of the farmland site and the soil acid-base value for comparison. In order to achieve the charac-

Table 1
Experimental conditions

Item	Device	Item	Server
A1. Data type	image jpg	B1. Network Storage	100 mb/per min
A2. Data size	180 kb/per image	B2. Image Analysis	600 file /per min
A3. Network Throughput	50 mb/per min		
A4. System Installation	local program installs (10 min)		
A5. Task execution	local setting (10 min)		
A6. Data collection times	[A2 * 30 (min) * 60 (1 hours) / A3] + A4 + A5		

Table 2
Single Device Experimental result

Case (One device)	1 hours (60 min)	2 hours (120 min)	3 hours (180 min)	4 hours (240 min)
File size (mb)	316.40625	632.8125	949.21875	1265.625
Network cost (times / min) upload	6.328125	12.65625	18.984375	25.3125
System Installation (times / min)	10	10	10	10
Job Setting (times / min)	10	10	10	10
Network cost Server Image Storage (times / min) download	3.1640625	6.328125	9.4921875	12.65625
Image Analysis (times / min)	3	6	9	12
Total Cost (times / min)	32.4921875	44.984375	57.4765625	69.96875
Total Storage File size (mb)	316.40625	632.8125	949.21875	1265.625

Table 3
Experimental conditions

Case (One device)	1 hours (60 min)	2 hours (120 min)	3 hours (180 min)	4 hours (240 min)
File size (mb)	316.40625	632.8125	949.21875	1265.625
Network cost (min) upload	1.265625	2.53125	3.796875	5.0625
System Installation (min)	10	10	10	10
Job Setting (min)	10	10	10	10
Network cost Server Image Storage (min) download	0.6328125	1.265625	1.8984375	2.53125
Image Analysis (times / min)	3	6	9	12
Total Cost (times / min)	24.8984375	29.796875	34.6953125	39.59375
Total Storage File size (mb)	63.28125	126.5625	189.84375	253.125

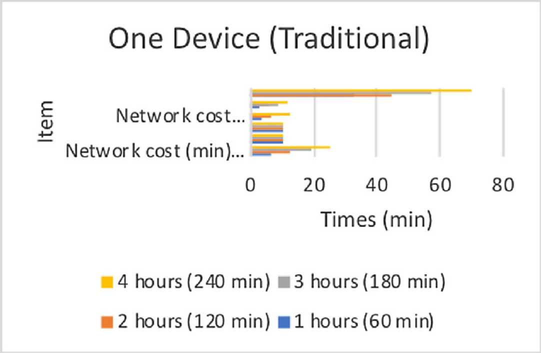


Fig. 14. Single Device with traditional data analysis.

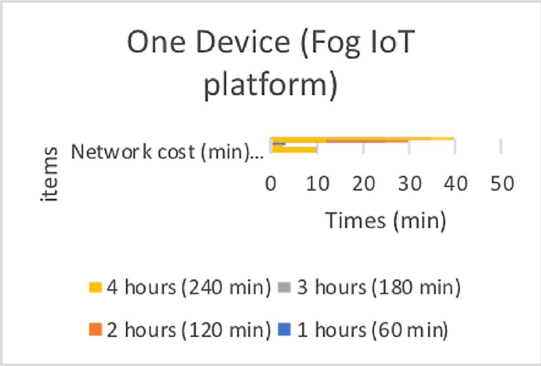


Fig. 15. Single Device with IoT Platform data analysis.

teristics of the innovative IoT service platform, the design sensing device in the research is the fog end link to automatically start the liquid fertilizer and the water supply. When detecting the suspected disease characteristics, the motor can compare the amount of fertilizer needed to improve the soil moisture data with the temperature data, and the fog equipment automatically informs each other to start the automatic water supply or nutrient mechanism in Figs. 16 and 17 .

7. Conclusions and future work

Through experimental simulation, the innovative status indicators were proposed by the innovative IoT platform, including the use of decentralised calculations to accelerate the verification of real-time computing results. The use of multiple Raspberry Pi devices creates a simulated decentralised environment and how to acquire different device sensing devices. to complete the simulated neural reflex behavior model, to provide improved empirical network model cannot achieve real-time operations. At the same time, in this study, the data analysis was performed using decentralised calculations. The analysis results were then transmitted over the network. The verification of experimental results can reduce the cost of network transmission. All data are analysed through pre-calculation, and the final analysis results will have value through data transmission. The post-analysis data is sent back to the platform environment. Other derived value-added service application systems can use the intermediate software to directly exchange information with the analysis results, which can reduce the cost of direct connection to the networked terminal equipment.

The research successfully validated an innovative networking service architecture and thinking method, providing a computer-based approach to creative services. These contributions can

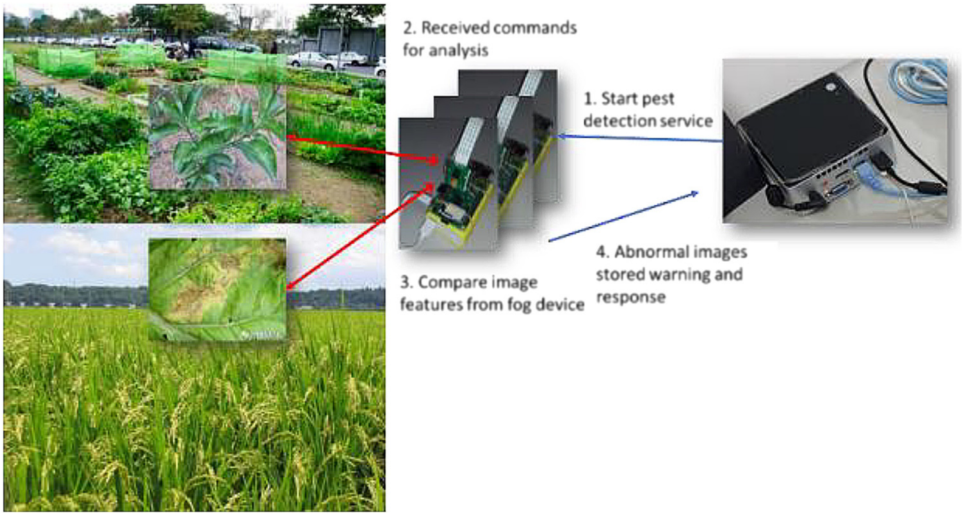


Fig. 16. The scenario of crop pest control.

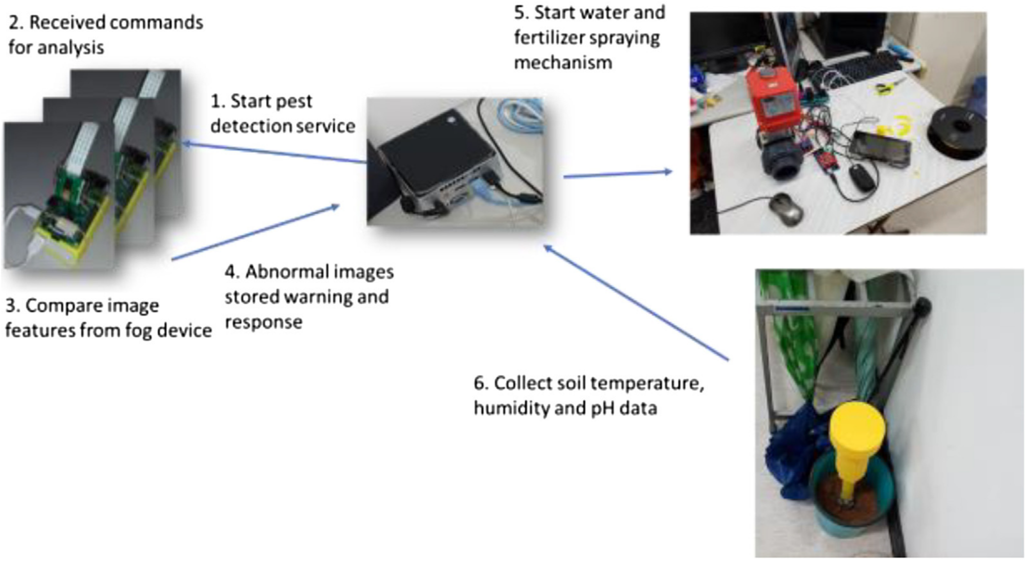


Fig. 17. Different IoT Devices Applied to Pest Control.

provide an integrated service and application of augmentation networking systems. In the research, the combination of cloud-side and fog-side systems can be used to establish fog computing models for different hardware devices through fog systems, allowing the device to have smart thinking and analysis capabilities. Establishing a creative service method can improve the following service indicators, including upgrading. Hardware computational efficiency, reducing computational costs by decentralised calculations; accelerating calculations to reflect response times, analysing and reacting results from a single centralised recovery, improving results from decentralised computations, reducing network service costs, and returning useful data only, Reduce the space capacity required for cloud storage.

Through the case study of the innovative networked service prototype development, it is proved that the proposed method can be innovative and can achieve the above three indicators, improving the dilemma of passive management of networked operations, and in the future, different field models can be derived for experimentation.

References

[1] E.T.S. Guilherme Pauli, An Analysis on the Use of IoT Devices in the Integrated Energy Resources Planning, in: 2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE), 2018, pp. 1–4.
[2] J. Wan, M. Chen, F. Xia, D. Li, K. Zhou, From machine-to-machine communications towards cyber-physical systems, Comput. Sci. Inf. Syst. 10 (2013) 1105–1128.
[3] J.L. Jaewoo Kim, Jaeho Kim, Jaeseok Yun, M2M Service Platforms: Survey, Issues, and Enabling Technologies, IEEE Commun. Surv. Tutorials 16 (1) (2013) 61–76.
[4] (a) J.M.S. Michael Vögler, Christian Inzinger, Schahram Dustdar, DIANE-dynamic IoT application deploymen, in: 2015 IEEE International Conference on Mobile Services, IEEE: New York, NY, USA, 2015, pp. 298–305; (b) Maher Abdelshkour, IoT, from Cloud to Fog Computing, Cisco Blogs, 2015.
[5] Wu-Chun Chung, Po-Chi Shih, Kuan-Chou Lai, Kuan-Ching Li, Che-Rung Lee, Jerry Chou, Ching-Hsien Hsu, Yeh-Ching Chung, Taiwan UniCloud: A Cloud Testbed with Collaborative Cloud Services, Proceedings of IEEE International Conference on Cloud Engineering (IC2E) (2014) 107–116.
[6] OM2M, Open Source platform for M2M communication http://www.eclipse.org/om2m/.
[7] Lidong Zhang, Yongwei Wu, Ruini Xue, Tse-Chuan Hsu, Hongji Yang, Yeh-Ching Chung, HybridFS - A High Performance and Balanced File System Framework with Multiple Distributed File Systems, To appear in Proceedings of IEEE Computer Software and Applications Conference (COMPSAC) (2017).

[8] Tse-Chuan Hsu, Dong-Meau Chang, Chih-Hung Chang, Rei-Heng Cheng, Rapid Construction of a Big Data Analysis System with Creative Comparison Methods, ISPAN-FCST-ISCC, 2017, pp. 485–490.

[9] “OSGi Specifications” Available: https://www.osgi.org/developer/specifications/.

[10] M. Chen, J. Wan, S. González-Valenzuela, et al., A Survey of Recent Developments in Home M2M Networks[J], IEEE Commun. Surv. Tutorials 16 (1) (2014) 98–114.

[11] C.L. Wu, C.F. Liao, L.C. Fu, Service-oriented smart-home architecture based on OSGi and mobile-agent technology[J], IEEE Trans. Syst. Man Cybernet. Part C (Appl. Rev.) 37 (2) (2007) 193–205.

[12] European Telecommunication Standard Institute, Machine-to-Machine Communications (M2M); mIa, dIa and mId interfaces, Available:, 2013 http://www.etsi.org/deliver/etsi_ts/102900_102999/102921/02.01.01_60/ts_102921v020101p.pdf.

[13] Jaewoo Kim, et al., M2M Service Platforms: Survey, Issues, and Enabling Technologies, IEEE Commun. Surv. Tutorials (2014).

[14] Z. Ji, I. Ganchev, M. O’Droma, L. Zhao, X. Zhang, A cloud-based car parking middleware for IoT-based smart cities: design and implementation, Sensors 14 (12) (2014) 22372–22393.

[15] A.J. Jara, A.C. Olivieri, Y. Bocchi, M. Jung, W. Kastner, A.F. Skarmeta, Semantic web of things: an analysis of the application semantics for the IoT moving towards the IoT convergence, Int. J. Web Grid Serv. 10 (2-3) (2014) 244–272.

[16] J. Gubbi, R. Buyya, S. Marusica, M. Palaniswamia, Internet of Things (IoT): A vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660.

[17] P.K. Gupta, B.T. Maharaj, R. Malekian, A novel and secure IoT based cloud centric architecture to perform predictive analysis of users activities in sustainable health centres, Multimedia Tools Appl. 2016 (2016) 1–24.

[18] D. Boswarthick, O. Elloumi, Olivier Hersent, M2M Communications: A Systems Approach, 2013, pp. 124–139.

[19] Sharief M.A. Oteafy, Hossam S. Hassanein, Towards a global IoT: Resource Re-utilization in WSNs, in: IEEE International Conference on Computing, Networking and Communications Invited Position Paper Track, 2012, pp. 617–622.

[20] Gregor Broll, Massimo Paolucci, Matthias Wagner, Perci: Pervasive Service Interaction with the Internet of Things, Internet IEEE Internet Comput. (2009) 74–81.

[21] Cheng Ru Lin, et al., “System and Method of Remotely Controllable Home Network System for Domestic Appliances”, Patent TW I319945, 2006.