ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 146

2nd International Conference on Artificial Intelligence: Technologies and Applications (ICAITA 2018)

An Architecture of IoT Application Support System in Launch Site

Litian Xiao[1,2,*], Mengyuan Li[1] and Nan Xiao[3]

[1]Beijing Special Engineering Design and Research Institute, Beijing 10028, China
[2]School of Software, TNList, KLISS, Tsinghua University, Beijing 10084, China
[3]China Aerospace Academy of System Science and Engineering, Beijing 10048, China
[*]Corresponding author

***Abstract*—Internet of Things (IoT) application support system is key infrastructures and common software platform for various IoT applications at launch sites. The research studies the progress of IoT middleware related to academia and industry. On the basis of hybrid architecture, the paper proposes the architecture of the IoT application support system for the launch site. The architecture combines software with hardware and includes design and implementation scheme. It contributes to solving the problems of IoT application support system about the heterogeneity of embedded sensor devices and mobile terminals, mobility and network transformation, communication and data exchange, a fusion of different middleware systems, and implementing middleware support environment in launch site. In the domestic operating system and database environment, the personnel positioning application is realized by graphical programming and application deployment. The actual example verifies the availability of the architecture and presents the developing process of the IoT application support system in the launch site.**

Keywords—Intenet of Things; application support system; architecture system; hybrid middleware; launch site

I. Introduction

To carry out comprehensive, long-term and reliable on-line monitoring for the equipment and environment of the launch site, it needs the relevant parameters of launch systems and flight articles to effectively collect, record, and analyze. The goal is realizing intelligent identification, positioning, tracking, monitoring and management of personnel and materials for the aerospace equipment and facilities.

Internet of Things (IoT) application support system is key infrastructures and common software platform of various IoT applications in launch sites. It integrates all operation applications to achieve the comprehensive application of resource sharing, data fusion, and intelligent decision-making. The IoT application support system is essentially a middleware between different platforms and application layer [1] [2]. The different platforms include servers, PC terminals, mobile terminals, embedded sensors, etc. The hierarchy diagram is shown in FIGURE I where MT is a mobile terminal and NFC is near field communication.

On the one hand, the support system is responsible for shielding the heterogeneity of the hardware devices, network and service platform for the lower. It makes the devices and equipment of launch site to be unified access and centralized management. On the other hand, it supports application developing, runtime sharing and open interoperability for the upper. And it guarantees the reliable deployment and management of IoT applications [3]. Therefore, the applications support system is the core and the foundation of IoT construction at the launch site. We propose the architecture of IoT application support system for the launch site. An experiment is carried out with the architecture for test application and has achieved good results.

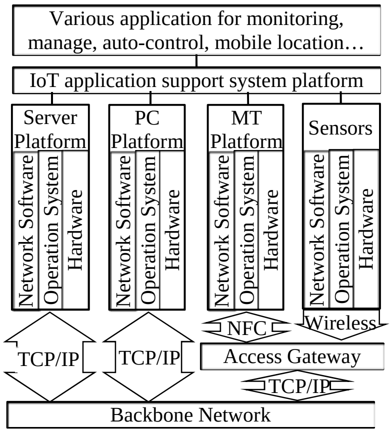

FIGURE I. HIERARCHY OF IOT APPLICATION SUPPORT SYSTEM IN LAUNCH SITE

II. Related Works

Based on the operation level, the application support system can be divided into application service middleware, embedded middleware, and hybrid middleware.

The application service middleware is usually deployed on the application server based on TCP/IP networks. It communicates with different sensor devices by loading drivers or adapters, and no additional hardware is required. The server is directly connected to the sensor devices. The middleware converges the sensory data of different devices into the event data needed by IoT applications. Typical application service middleware includes Oracle RFID middleware products [4] [5] and Rifidi [6] [7] simulation development platform which is open source. Its main advantage is easy to deploy. The disadvantages are two sides. One side is hardly conducive to distributed deployment because of the direct connection between the servers and the sensor devices. Another side is easy to single point failure because the sensor data is centrally processed by the servers.

The embedded middleware makes use of sensor devices to quickly filter and convert data or other preprocessing. Data processing is close to the source of the data so that network traffic and the stress of the application server can be reduced. The middleware mainly is wireless sensor network (WSN) middleware such as TinyDB [8] [9] and MetaQ [10]. The main advantages of embedded middleware are that sensor nodes bear data processing capacity and reduce the stress on networks and servers. The main drawbacks are limited services which the middleware only supports simple services, e.g. node discovery, data filtering and processing and code migration. The middleware also does not support heavyweight computing and storage services.

The hybrid middleware divides the application service middleware into two parts. The one part is the common service and runtime support environment. Another part is the sensing device abstraction layer and runtime support environment. The former is deployed on the server. The latter is deployed in a special access gateway. The more representative hybrid middleware is the IBM RFID middleware products [11] [12] and the SYBASE RFID middleware products [13] [14].

The hybrid middleware is more suitable for the IoT application support system in launch site, but there are a large number of heterogeneous embedded sensing devices and the introduction of mobile terminals. It is necessary to solve the problems of IoT application support system in launch site from architecture. We address oneself to architecture construction for the followings:

- Heterogeneity of embedded sensor devices and mobile terminals.
- Mobility and network transformation.
- Communication and data exchange.
- Fusion of different middleware systems.
- Implementation of middleware support environment for IoT.

III. Architecture Design of IoT Application Support System for Launch Site

The IoT of launch site should provide the launch capability required to enable exploration while reducing operations and maintenance costs, and achieve a reduction in ground safety mishaps, process escapes, and close calls. It is designed to adopt the technology architecture of componentization, service, and networking. It mainly consists of seven parts including basic software and hardware environments, databases, common services, application components, integrated portals, developing tools, standards, and specifications, etc., as shown in Figure II.

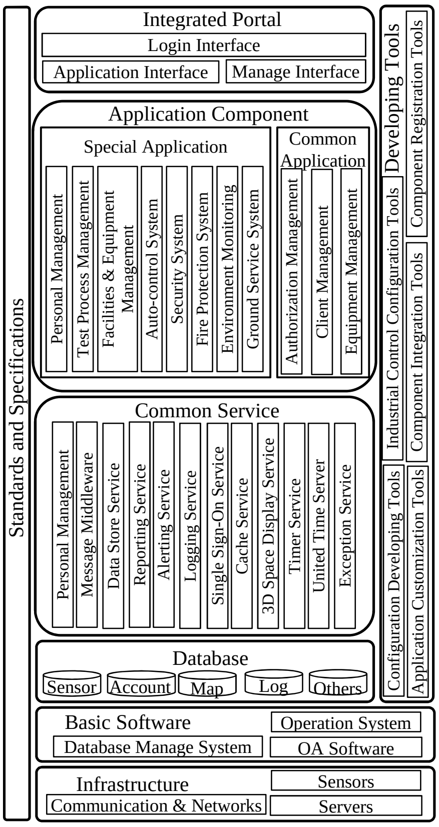

FIGURE II. ARCHITECTURE OF IOT APPLICATION SUPPORT SYSTEM AT LAUNCH SITE

A. *Basic Software and Hardware Environment*

It mainly includes hardware and operation system such as networks, computers, sensors, and software environments such as database management system and office software.

B. *Database*

The database is composed of sensor database, account database, map database and log database and so on.

C. *Common Service*

Common service consists of access service, message middleware, alarm service, single sign-on service, 3D space display service, united time service, data storage service and so on.

D. *Application Components*

It needs to be integrated automatic control system, security system, fire protection system, environmental monitoring, and exception application system. Also, the integrated components still include ground personnel control, test process management, facility management and other new applications. To perfect the management system, it provides privilege management, client management, equipment management and other public applications.

The component of operating environment provides the container for the application component operating, which enables the application component to work in the application system platform. Component operation support environment provides basic support for component operation mainly including event service, data service, network service, protocol-driven service and data exchange service.

Application integration platform provides application registration tools, application integration tools, and application customization tools. Through the platform, users can flexibly combine the applications developed to customize the data flow interaction between applications. It completes business, process, and user usage function on the application platform.

E. Integrated Portal

It provides users with a unified single sign-on login. User login can customize their own business applications through the integrated portal. It is integrated with the application system operation platform and environment.

By loading the output configuration of the integrated development platform, the integrated operating environment integrates business applications, user permissions and rules of data interaction to provide the user with the final operation.

F. Developing Tools

It is consists of the development environment, application publish, component management and so on. Application development platform is based on domestic operating system and database, industrial control configuration tool, etc.

G. Standards and Specifications

It specifies the interface specification and integrated application development specification between the modules of the system.

Through common service, the architecture is suitable for the heterogeneity of embedded sensor devices and mobile terminals because of hardware devices, operating systems and network protocols produced by different manufacturers. It can coordinate the capacity differences in storage, computing, and communication.

The architecture makes network environment keep the stability when various intelligent terminal devices access (or exit) the IoT at the launch site, or mobile locating. The data types and data access control methods are different in the wired backbone network and the wireless sensor access network. The network communication service combines multiple communication modes based on different application requirements. Forming a set of communication and information exchange standards under the architecture, a communication mechanism is designed to support the access and exchange of various types of data.

The traditional application service middleware based on server platform has been developed to provide data storage, numerical calculation, statistics mining, policy control, TCP/IP communication services and other common services [15] [16] [17]. Because the features of sensor nodes are mass heterogeneous data transmission, the storage and forwarding functions including fusing data and computing are deployed in embedded middleware to reduce the cost of application server and backbone network in the architecture.

All application components in the IoT of launch site can be customized through the auxiliary tools of the system. After each component is developed, it can operate and complete a specific business independently. It can be integrated through the development tools of the system to complete the data exchange among the individual components. Thus, the business applications are not only independent of each other in their business functions, but also interconnected in the data flow. In this way, the IoT in the launch site can be constructed by blocks. The approach can reduce the coupling within the system and improve the system expansibility and maintainability.

IV. Application Development and Deployment Cases

As an example, a simple case introduces the application flow of the development and deployment of the architecture of IoT application support system in the launch site for personnel location.

The case is that the position of the personnel in the room is displayed in real time. It utilizes the ZigBee anchor node and positioning tag to realize the personnel location located in the room.

The application mode of IoT application support system adopts the trinity mode of custom-integration-operation. A special application is custom-developed by auxiliary development tools. It is integrated the applications and built into the system by auxiliary integration tool. The integrated portal makes system apply and business operate. The mode is shown in Figure III.

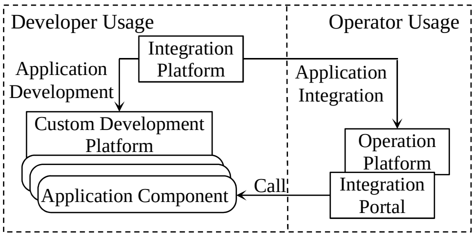

FIGURE III. MODE DIAGRAM OF SYSTEM APPLICATION

Its positioning principle is described as following. The anchor node is fixed the coordinate in the room. The positioning tag card sends the wireless signal to the anchor node in real time. And the anchor node sends the received signal to the location service component. The component converts the field strength of the received signal to get the distance between the tag card and each anchor node. Then it calculates the coordinates of the positioning tag card by using the trilateral measurement algorithm combined with the coordinate of the anchor node. Main sensing devices are 3 ZigBee anchor nodes and 4 positioning tag cards. Personnel location application has 6 steps for development and deployment.

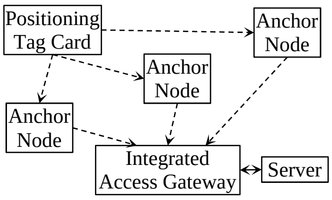

FIGURE IV. CONNECT RELATION OF EQUIPMENT HARDWARE

A. Device Connection

As shown in Figure IV, three ZigBee nodes are wirelessly connected to the positioning tag card and the integrated access gateway via ZigBee. The integrated access gateway connects to the server via a network cable.

B. Driver Development

The component development tool is used to write the driver component of ZigBee anchor node protocol loaded into the integrated access gateway.

C. Service Component Development

The component development tool is used to write positioning service components loaded into the integrated access gateway.

D. Positioning Logic Design

As shown in Figure V, the data of Zigbee anchor node is abstracted into a field strength data component. It connects the output of the field strength data component with the input of the location service component. The output of the location service component is connected to new computing location result component. Then the logical design of the location application is completed, and location coordinate is generated by the sensing data through the flow direction from left to right.

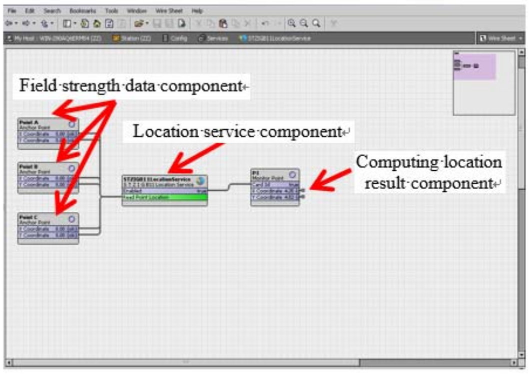
Field strength data component
Location service component
Computing location result component

FIGURE V. GRAPHIC PROGRAMMING

E. Interface Design

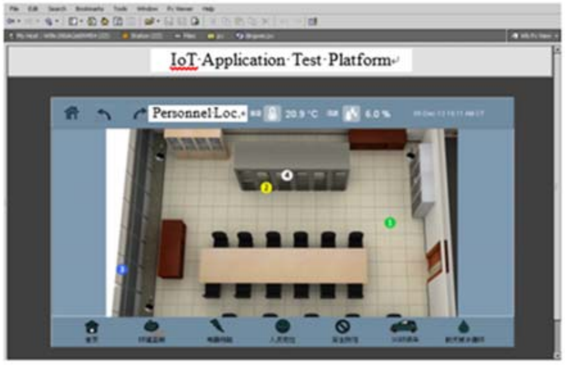

FIGURE VI. APPLICATION INTERFACE OF PERSONNEL LOCATION

As shown in Figure VI, anchor node icons and positioning tag card icons are added from 1 to 4 on the floor of the room plan.

F. Application Deployment

The computing location result component is associated with the location tag card icons, and the positions of the location icons in the FIGURE VI are changed with the coordinates. So far, the application deployment is completed.

V. Conclusion

The research studies the progress of IoT middleware related to academia and industry. The challenges and existing problems are analyzed for application in launch site. On the basis of hybrid architecture, the paper proposes the architecture of the IoT application support system for the launch site. The architecture combines software with hardware and includes design and implementation scheme. In the domestic operating system and database environment, the personnel positioning application is realized by graphical programming and application deployment. The real example verifies the availability of the architecture and presents the developing process of the IoT application support system in the launch site.

Next, we shall further study the relevant standards and specifications which lag has affected the application. There are many different ports to the components in the architecture. A lot of converting work on interactive ports needs to be done in the actual application. Rapid application and deployment can be promoted under the unified requirements of the architecture. Then it can improve the abilities and efficiency of the launch site.

Acknowledgment

This work was funded by the project of General Technology on Test and Launch partially sponsored by the 973 program. We would like to thank Tsinghua's Prof. Yu Liu who provided us with the helpful suggestions and some research resources in the research.

References

[1] S. Huda, A. Wesam, E.S. Khair, "Internet of Things: A Review to Support IoT Architecture's Design," IT-DREPS Conference, Amman, Jordan, Dec 6-8, 2017.

[2] S. Li, L.D. Xu, S. Zhao, "The internet of things: a survey," Information Systems Frontiers, 2015, vol.17(2), pp.243-259.

[3] J. Lin, W. Yu, N. Zhang, X. Yang, et al., "A survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and applications," IEEE Internet of Things Journal, 2017, vol. 99, pp.1-11.

[4] Oracle，http://www.oracle.com, 2018.

[5] N Deng, "RFID Technology and Network Construction in the Internet of Things," International Conference on Computer Science & Service System, 2012, pp.979-982.

[6] Rifidi，http://transcends.co, 2018.

[7] S.W. Zhou, Y.U. Guo, R. Zhang, C. Qin, "Based on RFID emulation in Rifidi environment," Electronic Design Engineering, 2011, vol. 5, pp.12-15.

[8] TinyDB, http://tinydb.readthedocs.org, 2018.

[9] P.D. Felice, M. Ianni, L. Pomante, "A spatial extension of TinyDB for wireless sensor networks," IEEE Symposium on Computers & Communications, 2008, pp.1076-1082.

[10] J. Liao, X. Zhuang, R. Fan, X. Peng, "Toward a General Distributed Messaging Framework for Online Transaction Processing Applications," IEEE Access, 2017, vol. 99, pp.1-11.

[11] IBM，http://www.ibm.com, 2018.

[12] A.M. Khan, A. Khaparde, V.P. Savanur, "Self-aware inventory system based on RFID, sensors and IBM security directory integrator,"

International Conference on Inventive Computation and Technologies, 2017, pp.1-4.

[13] SYBASE, http://infocenter.sybase.com, 2018.

[14] F. Wang, S. Liu, P. Liu, “Complex RFID event processing,” VLDB Journal, 2009, vol. 18(4), pp.913-931.

[15] V. Gazis, M. Görtz, M. Huber, et al., “A survey of technologies for the internet of things,” International Wireless Communications & Mobile Computing Conference, USA, Dec. 2015.

[16] M. Jose, “P2.42: Latest Development in Advanced Sensors at Kennedy Space Center (KSC),” IEEE Sensors, 2002, vol. 2(2), pp.1728-1733.

[17] M.V. Ramkumar, N.R. Prasad, R. Prasad, “Middleware Architecture for Next Generation Heterogeneous Networks,” Wireless Personal Communications, 2012, vol. 66 (3), pp.577-593.