Advanced EPC Network Architecture Based on Hardware Information Service

HAN Tianyu, ZHU Siyu, XIE Bin, and TIAN Juan

(China Academy of Information and Communications Technology, Beijing 100191, China)

Abstract: Object traceability based on Radio Frequency Identification (RFID) is an important capability of the Internet of Things (IoT). GS1 EPCglobal, a de-facto standard in RFID technology, develops Electronic Product Code (EPC) network architecture, which mainly includes EPC Information Service (EPCIS), Object Name Service (ONS), and Discovery Service (DS). This architecture is used to capture and share standardized events representing various aspects on EPC object. However, the EPC network architecture also faces challenges; for example, the separate management of unrelated event data and master data may increase time consumption when an application extracts valuable information by combing them. A Hardware-Based Information Service (HIS) device is raised in this paper, which is installed on the specific EPC tagged object directly and this could alleviate the problem above. The HIS could store the event data and master data related to the EPC identifier centrally and the application could gain the basic information based on the data stored in the HIS without utilizing ONS, DS and EPCIS.

Keywords: EPCIS; lifecycle information; hardware information service

DOI: 10.12142/ZTECOM.202001004

http://kns.cnki.net/kcms/detail/34.1294.TN.20200316.1731.010.html, published online March 17, 2020

Manuscript received: 2019-12-01

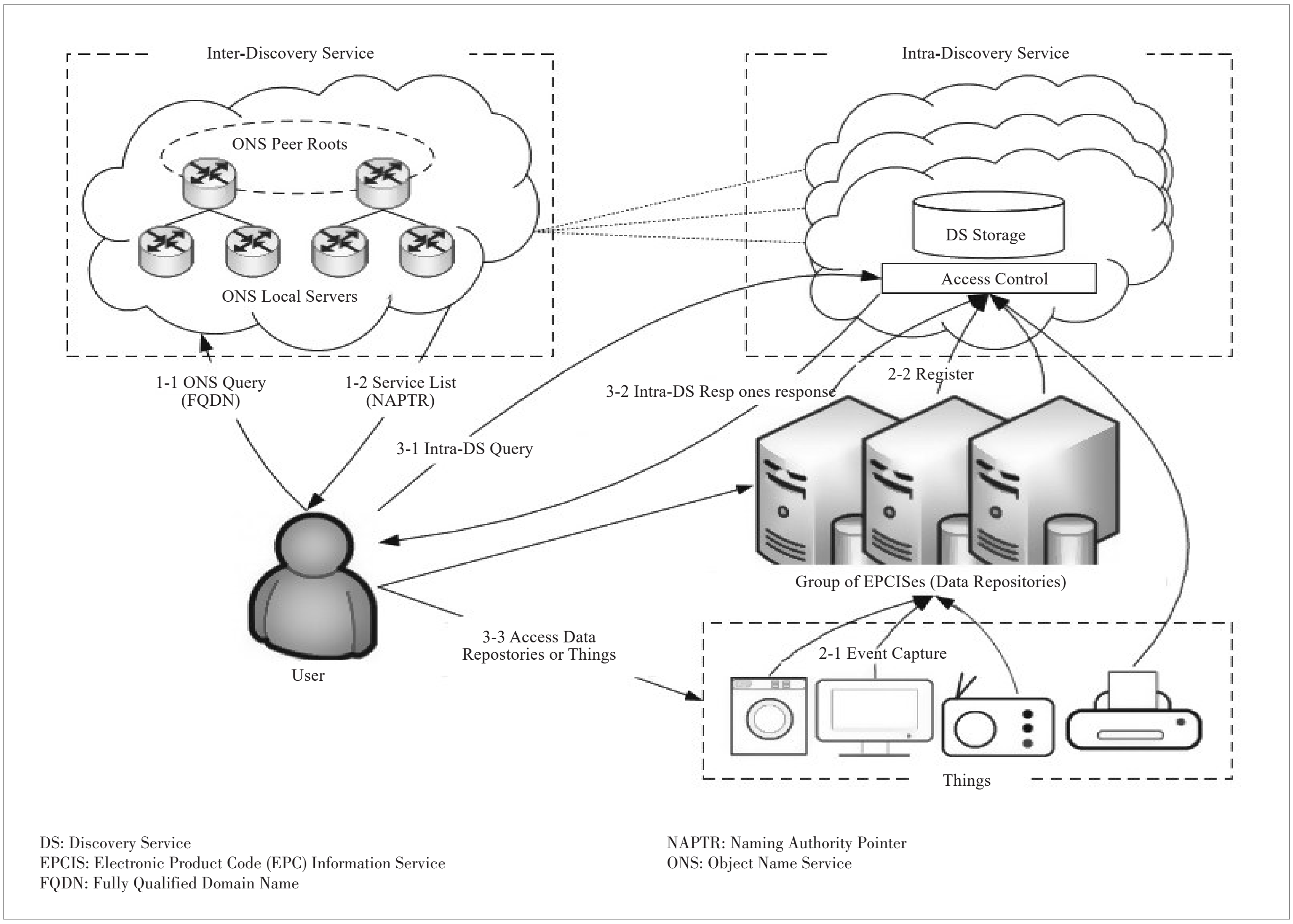
Citation (IEEE Format): T. Y. Han, S. Y. Zhu, B. Xie, et al., "Advanced EPC network architecture based on hardware information service," *ZTE Communications*, vol. 18, no. 1, pp. 18–23, Mar. 2020. doi: 10.12142/ZTECOM.202001004.

1 Introduction

The Electronic Product Code (EPC) network system is based on the Radio Frequency Identification (RFID) and Internet architecture. It assigns a globally unique encoding to each physical object. EPC technology system is considered as a scientific and authoritative identification system. EPC encodings are mainly aimed at improving the reliability and efficiency of supply chains. They are widely used in some open circumstances, for example, the smart agriculture and logistic. The EPC network system basically includes EPC identifier, EPC Information Service (EPCIS), Object Name Service (ONS), and Discovery Service (DS).

Fig. 1 shows one of the most widely used EPC network system [1]. In response to Fully Qualified Domain Name (FQDN) typed ONS query, ONS Returns Naming Authority Pointer (NAPTR) typed service list for the given object. Here, FQDN and NAPTR belong to the standard massage formats of Domain Name System (DNS). By adding end-point of DS as one of the service elements in ONS, user can access DS which actually locates the desired data repositories (Steps 1-1 and 1-2 in Fig. 1). DS has two data flows: write and read flows. The

This work was supported by the 2018 Industrial Internet Innovation and Development Project —— Industrial Internet Identification Resolution System: National Top-Level Node Construction Project (Phase I).



▲**Figure 1. Overview of discovery service.**

write flow occurs when the event is captured from things and the spatial-temporal data extracted from the event is registered to DS (Steps 2-1 and 2-2 in Fig. 1). The spatiotemporal data includes time and logical location of repositories for indirect access things' data through data repositories. It may also include physical location and logical location of things for finding the physical location and direct accessing things. There an access control mechanism is necessary to ensure the safety and privacy of the data in these steps. The read flow occurs when the users search the logical locations of desired repositories or things (Steps 3-1, 3-2, and 3-3 in Fig. 1). Similar with the write flow, before querying the main storage service, authority check is established. After the authority is verified, the users would receive a list of EPCIS logical addresses related to the query and the users could access each of the EPCIS repositories respectively to extract the information needed.

Based on the commonly used event-oriented EPC network structure, we address several possible problems as follows [2]:

1) EPCIS separately manages event data (i. e. , aggregation event, transaction event, etc.) and master data (i. e. , static information over the whole life of objects) so that the accessing application needs to combine them to obtain all the information on an object with additional cost.

2) The accessing application needs to search all the event types over again until there is no additional event found and extract target information from the reconstructed data since the accessing application may not know what event type is used for and what object is transformed from or transformed to the queried object in advance.

3) Under an event-oriented approach, each event can include a business context on one or more objects so that it may make EPCIS inefficient when an access control mechanism must be enforced on the selected objects. For example, the owner of a ranch may want to share business contexts on the cow only with the slaughterhouse owner. However, to enforce this kind of instance-level access control, a query interface of EPCIS becomes too complicated to filter them appropriately.

This paper proposes a Hardware-Based Information Service (HIS) device, which is mainly installed on some intelligent equipment or machines. The main idea of HIS is that it could centrally store the basic lifecycle information, which is used to be stored in the distributed EPCIS servers. Therefore, the client only needs to simply search the HIS to obtain the complete traceability information of the item. There are two application scenarios of HIS. One is to interact with EPCIS and the other is to communicate with other HISs. Communicating with EPCIS is to fit to the existing EPC networks for centrally storing the distributed data in EPCIS servers. Exchanging information among HISes is to record the events that occur between EPC objects. Thus, the detector in the HIS will identify the devices in the scene firstly in order to determine the mode of data interaction. Next, the event processor will process the data received from HIS or EPCIS and store it in the memory. Finally, users can access the data in HIS in a flexible way.

Although HIS can only store the basic information about EPC items, users can directly obtain the complete lifecycle data, so that they can make further targeted query according to the information provided by the HIS, instead of iteratively querying for the information utilizing discovery service. Meanwhile, with the use of HIS, the number of query to the EPC system could be reduced dramatically. Thus, the work load of the EPC network can be significantly alleviated.

It should be noticed that the implementation of HIS is based on the EPC network, which is no need to be modified to fit HIS. Although the HIS can gain the data of the item easily, the EPC network can still be used for further searching if the user is interested in the master data that is not stored in the HIS. It takes less time to access the master data because the user could use ONS to find the location of the related EPCIS without applying discovery service.

2 Related Work

For several years, many works for alleviating the problems mentioned above have been conducted. Semantic web technologies lead to the development of EPCIS variants. Linked EPCIS [3] – [5] presents an ontological model of EPCIS events to improve supply chain visibility. However, Linked EPCIS is not compatible with the standard capture interface and does not provide scalability for a large amount of event data. Unlike the event-oriented persistent approach of Linked EPCIS, EPC Graph is a graph-oriented persistent approach [1], [6]. EPC Graph establishes efficient and privacy-enhanced traceability by representing the EPCIS document as properties and relationships in/among objects and locations. However, this solution is not compatible with the EPCIS accessing application because it does not provide a standard query interface.

Some works have focused on how DS is efficient designed [7], [8]. Bridge DS, the most early presented architecture design, suggests eight different DS architectures. Among them, four efficient architectures are selected considering how fast the response latency is and easily the secured resource is protected. Some works focus on the performance issues to find one of distributed DSes. For this, most of them apply peer-to-peer (P2P) technology to DS, Especially Distributed Hash Table (DHT) [9] – [11]. DHT offers high robustness, single failure avoidance, and load distribution. However, the previous works do not seriously deal with the intra-DS aspect. Any deterioration in the intra-DS may cause great performance degradation of entire DS. Additionally, security issues are also important for protecting secured resources in DS.

3 Hardware Information Service Device

3.1 Architecture of HIS

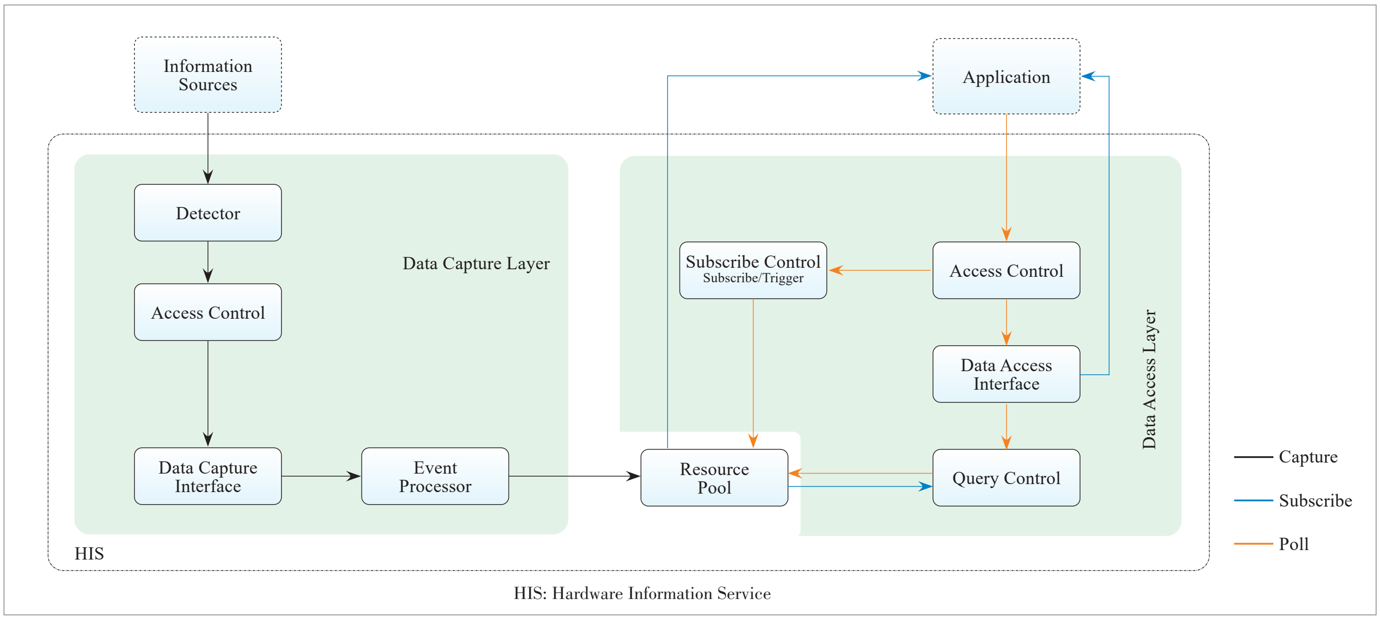
To improve the performance of the current EPC network system, we propose a HIS device, which is installed on the EPC-tagged item itself. The HIS of an item stores all the basic data (event data and master data) related to the EPC identifier and the device could be any small equipment like the Universal Serial Bus (USB) disk that we daily use.

Items installed with HIS need to meet certain conditions. HIS is mainly used in self-powered devices, which can search for their own energy supply; for example, the self-driving car that can intelligently find its own charging pile, or the robot that can generate electricity by utilizing other kinds of energy. Self-powered devices usually have mobility and intelligence. When self-powered equipment is running, people usually do not know its location or what happened to it. In this way, a certain device is needed to record their behavior. However, the things that are not self-powered such as foods and drugs do not need equipping with HIS because the limit amount of dynamic data generated by this kind of things could be processed by the existing EPC network system easily and the most queries related to them ask for the static data without distributed processing involved, which makes the response relatively faster.

As illustrated in **Fig. 2**, HIS could be separated into three parts according to their functions: the data capture layer, resource pool, and data access layer. The data capture layer is used to capture data (event and master data) and store it into the resource pool. The data access layer is implemented to publish and access data.

3.2 Procedure of Implementing HIS

Algorithm 1 shows the specific operation process of the data capture layer. The detector is used to detect the interactive device in the scene and broadcast permission and device type related information. The access control is implemented to get the device type and the state code of the device detected. The state code is introduced because of the safety and privacy concerns. The data capture interface is utilized to receive standardized both dynamic and static data from the device connected. After that, the event processor is needed to handle the data properly. Actions like data cleaning, sorting the events by the EventTime, and filtering unreasonable data should be taken in order to capture the valuable information and store it in the resource pool, which could be regarded as the memory of HIS.



▲**Figure 2. The structure of HIS.**

Algorithm 1. Data Capture Layer.

```
procedure CaptureData
    // Capture the data from EPCIS or HIS
    EPC: the identifier of the object
    L: the broadcast massage received
    U: the data stored in the HIS
    while ture do
        // broadcast the permission information and the de-
          vice type
        Broadcast;
        if device detected then // privacy protection
            state_code, type = permission_check(L);
            if state_code is 2 then
                connection established;
                send(U);
            else if state_code is 1 then
                if type is EPCIS: then
                    Interact_EPCIS(EPC);
                end if
                if type is HIS: then
                    Interact_HIS;
                end if
            else
                connection rejected;
            end if
        end if
    end while
end procedure
procedure Interact_EPCIS(EPC)
    R1: the date related to the EPC stored in the EPCIS
    connection established;
    R1 = query(EPC);
    U = store(R1);
    // data cleaning, sorting by the EvenTime, etc.
    process(U);
end procedure
procedure Interact_HIS
    R2: the data generated by the EPC object itself
    connection established;
    R2 = generater;
    U = store(R2);
    process (U);
end procedure
```

In the data access layer, there are many types of HIS requests, which could be similar with the requests of EPCIS [12]. Users can obtain the data from the repository immediately through the poll method. Meanwhile, event data can be delivered to a specific destination periodically (i. e. scheduled subscription) or if a specific condition meets at the time of capture (i. e. trigger subscription) via the subscribe method. The former subscription is useful for automating event delivers while the later one is suitable for developing real-time monitoring applications. Both of the subscription methods store their configurations in the resource pool. The access control is introduced to protect the privacy and enhance the safety of the system. Subscribe control and query control are required to manage subscribe and poll methods respectively.

3.3 Application Scenarios of HIS

There are two main application scenarios. One is between HIS and EPCIS and the other is between HISes.

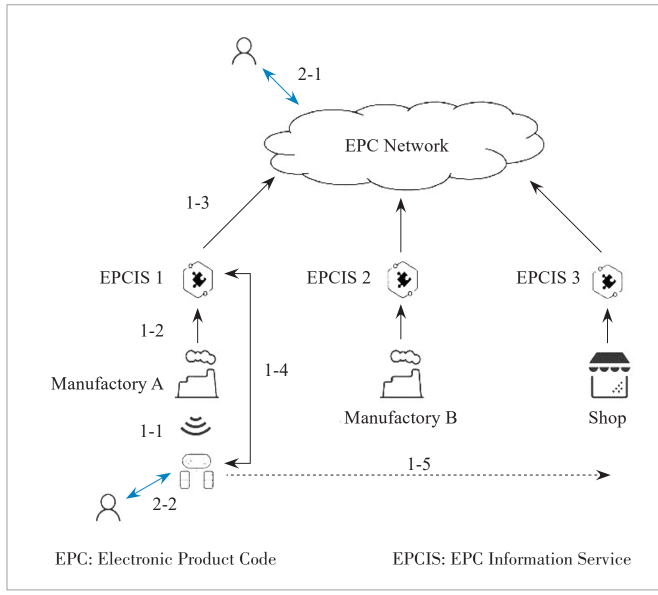
For the former scenario (**Fig. 3**), in the data capture phase, when a self-powered item enters the scene, the factory will generate corresponding data and send it to the EPCIS of the factory (Steps 1-1 and 1-2 in Fig. 3). The EPCIS of the factory is registered to the EPC network system in advance, so that the data of the manufactory could be found by the EPC network (Step 1-3 in Fig. 3). When EPCIS obtains data, it will synchronize it to the HIS (Step 1-4 in Fig. 3), so that the complete lifecycle could be maintained in HIS. The processes above would be repeated in the lifecycle of the item when entering the similar scenes (Step 1-5 in Fig. 3). In the data access phase, users can obtain data through the EPC network (Step 2-1 in Fig. 3) or access to HIS directly (Step 2-2 in Fig. 3).

For the later scenario (**Fig. 4**), self-powered devices generate the data themselves and interact with each other through HIS.

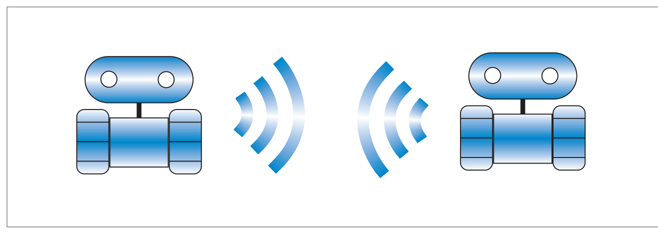
3.4 Evaluation of HIS

In order to compare the difference between the time required for the EPC network query method and the HIS query, we abstractly divided the time into t1 – t8. As shown in **Fig. 5**, the query method of the EPC network consists of all the time phases: t1 is the time phase of the discovery server receiving the request to retrieve the network address of the EPCIS servers related; t2 is the time period when the ONS searches the discovery service related to the EPC object or the network address of the manufacturer's EPCIS server; t3 and t4 are the time periods for the EPCIS server of the manufacturers and the participators to retrieve the relevant data respectively; t5 is the time period for transmitting the data retrieved to the discovery server; t6 is the time period for transmitting the data retrieved by the EPCIS of manufacturers to the user; t7 is the time period for transmitting the data aggregated in the discovery server to the user; t8 is the time period for the user to reconstruct the returned data and extract meaningful information. For HIS, since the basic information of the article is already stored, it only needs to query the master data of interest by utilizing ONS. **Table 1** shows the query time between the EPCIS and the HIS. It can be seen that the query efficiency of HIS will be relatively high.

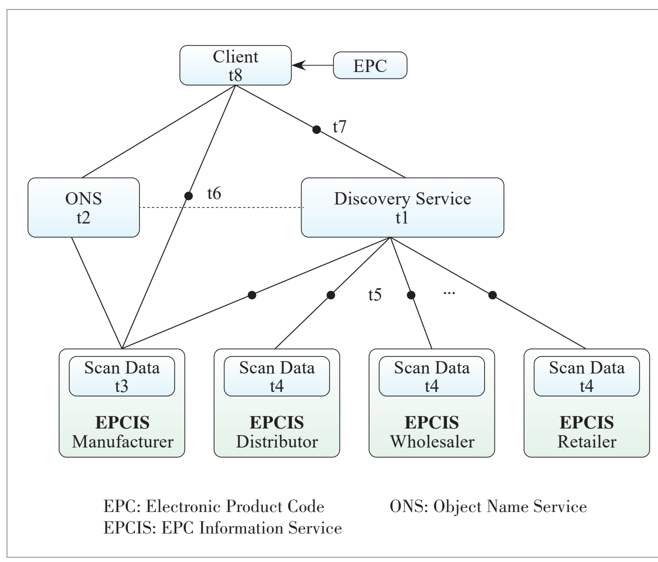
HIS have multiple advantages. Firstly, HIS stores all the basic data related to an item. In this way, people can gain the target information easier and faster than the tradition approach because no discovery service query is needed and the accessing application can easily indicate the relationship between the events. Secondly, because the client could access HIS directly, the load of the EPC network can be reduced significantly. Finally, HIS is designed for the future usage. The hardware can be installed on the robots, self-driving cars, drones and so on for monitoring, predicting and describing. In addition, users can access the information regardless of time, place and environment.



▲Figure 3. The Hardware Information Service (HIS)-EPCIS scenario.



▲Figure 4. The Hardware Information Service (HIS)-HIS scenario.



▲Figure 5. The EPC network query process.

▼Table 1. The query time required for the EPC network and HIS

Method	t1	t2	t3	t4	t5	t6	t7	t8
EPC network	●	●	●	●	●	●	●	●
HIS		●	●			●		

EPC: Electronic Product Code HIS: Hardware Information Service

4 Conclusions

In this paper, the potential issues of the currently widely used EPC network are evaluated. The issues include the inefficiency of the EPC network led by the distributed storage of the unrelated data and the access control mechanism of discovery service. Thus, the hardware information service is proposed to alleviate the challenges and improve the performance of the system. The basic idea of HIS is that it centrally stores the basic lifecycle information of the attached object and the user accesses it directly without involving discovery service, which may decrease the response time and alleviate the work load of EPC network. The device is specifically designed for the self-powered equipment. It has two different modes of data interaction: One is to exchange information with EPCISs in the existing network and the other is to communicate with other objects equipped with HIS. Both of the methods are used to capture the data from where it generated. Though HIS may improve the performance of system, the cost of implementing such mechanism is relatively high and the safety and robustness of EPC network still remain unsolved. For the further work, we will construct more realistic experiment environments by deploying IoT devices and getting real-data from them. Additionally, we will research on the searching performance improvement of the EPC network.

References

[1] KWON K, KIM D, KIM D. Oliot-Discovery Service: Dealing with Performance and Security Issues from Intra-DS Aspect for IoT [C]//IEEE Global Communications Conference (GLOBECOM), Washington DC, USA, 2016. DOI: 10.1109/glocom.2016.7842037
[2] BYUN J, KIM D. EPC Graph Information Service [M]//Lecture Notes in Computer Science. Cham, Switzerland: Springer International Publishing, 2015: 232 – 246.DOI: 10.1007/978-3-319-26190-4_16
[3] SOLANKI M, BREWSTER C. Representing Supply Chain Events on the Web of Data [C]//Detection, Representation, and Exploitation of Events in the Semantic Web, Workshop in Conjunction with the 12th International Semantic Web Conference. Sydney, Australia, 2013: 18 – 29
[4] SOLANKI M, BREWSTER C. EPCIS Event-Based Traceability in Pharmaceutical Supply Chains via Automated Generation of Linked Pedigrees [M]//The Semantic Web – ISWC 2014. Cham: Springer International Publishing, 2014: 82 – 97.DOI: 10.1007/978-3-319-11964-9_6
[5] SOLANKI M, BREWSTER C. Modelling and Linking Transformations in EPCIS Governing Supply Chain Business Processes [M]//Lecture Notes in Business Information Processing. Cham, Switzerland: Springer International Publishing, 2014: 46 – 57.DOI: 10.1007/978-3-319-10491-1_5
[6] BYUN J, WOO S, KIM D. Efficient and Privacy-Enhanced Object Traceability Based on Unified and Linked EPCIS Events [J]. Computers in Industry, 2017, 89: 35 – 49. DOI: 10.1016/j.compind.2017.04.001
[7] EVDOKIMOV S, FABIAN B, KUNZ S, et al. Comparison of Discovery Service Architectures for the Internet of Things [C]//IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing. Newport Beach, USA, 2010. DOI: 10.1109/sutc.2010.22
[8] BRIDGE. High Level Design for Discovery Services [EB/OL]. (2007-08-31) [2019-11-15]. http://www.bridge-project.eu
[9] LORENZ M, MUELLER J, SCHAPRANOW M P, et al. A Distributed EPC Discovery Service Based on Peer-to-Peer Technology [C]//7th European Workshop on Smart Objects: Systems, Technologies and Applications; RFID SysTech 2011.Dresden, Germany, 2011: 1 – 7
[10] LIU P, KONG N, TIAN Y, et al. A Distributed EPC Discovery Service Based on Peer-to-Peer Technology [C]//IEEE International Conference on Internet of Things (iThings). Taipei, China, 2014
[11] MANZANARES-LOPEZ P, MUÑOZ-GEA J P, MALGOSA-SANAHUJA J, et al. An Efficient Distributed Discovery Service for EPCglobal Network in Nested Package Scenarios [J]. Journal of Network and Computer Applications, 2011, 34(3): 925 – 937. DOI: 10.1016/j.jnca.2010.04.018
[12] BYUN J, WOO S, TOLCHA Y, et al. Oliot EPCIS: Engineering a Web Information System Complying with EPC Information Services Standard towards the Internet of Things [J]. Computers in Industry, 2018, 94: 82 – 97. DOI: 10.1016/j.compind.2017.10.004

Biographies

HAN Tianyu received the B. E. degree in automation from Beijing Jiaotong University, China in 2016, and the M.S. degree from the School of Computer Science, University of New South Wales, Australia, in 2019. He works at the industrial Internet and Internet of Things Institute, the China Academy of Information and Communications Technology. His research interests include the areas of industrial Internet and Internet of Things, especially the data model and data sharing mechanism.

ZHU Siyu received her Ph.D. degree in signal and information processing from the Chinese Academy of Sciences, China. She now works at the industrial Internet and Internet of Things Institute, the China Academy of Information and Communications Technology. Research areas include industrial Internet Identifier Resolution, Blockchain and Internet of Things.

XIE Bin received the B.E. degree in automation from Beijing Forestry University, China in 2012, and M.S. degree in control theory and control engineering from Beihang University, China in 2019. She is currently a research engineer of the Institute of industrial Internet of Things, the China Academy of Information and Communications Technology. Her research interests include industrial Internet data models and data sharing mechanism for the industrial Internet of Things.

TIAN Juan (tianjuan@caict.ac.cn) received the B.E. degree in electronic information engineering from Chengdu University of Information Technology, China in 2005, and M.S. degree in signal and information processing from Chengdu University of Information Technology in 2008. She is currently a research director of technical research development of the Institute of industrial Internet of Things, the China Academy of Information and Communications Technology. Her research interest includes the area of the industrial Internet with focuses on data acquisition and automatic identification technology, and network communication technology.