Temporal Data Modeling for RFID Data in the Retail Industry

Hong-won Yun and Jung-hwa Lee, *Member, KIMICS*

Abstract—RFID applications are mostly associated with the timestamp when the events occur. For managing RFID data we need to design data modeling for RFID data to support monitoring and temporal queries. In this paper, we propose a temporal RFID data modeling to maintain the history of events and state changes and to monitor the states of RFID objects. This data modeling involves essential basic operations for RFID data to monitor the information of RFID objects and to support temporal queries. Filter operations can achieve better query performance and obtain efficient storage space usage. It is possible to adapt into different temporal business applications.

Index Terms—RFID, Temporal data modeling, EPC, RFID applications, Temporal query

I. INTRODUCTION

RFID (Radio Frequency Identification) technology has received a great deal of attention over the past couple of years. It is being gradually adopted and deployed in a wide area of applications including supply chain and logistics, highway tolls, retail, and security [1,2]. Large retailers have already started implementing RFID systems in their warehouses and distribution centers. The technology holds the promise to streamline supply chain management, facilitate routing and distribution of products.

RFID technology is possible to create a physically linked world in which every object is identified, located, and tracked. It can achieve greater visibility and product velocity across supply chains, easier product tracking and monitoring. The diversity of RFID applications and the unique characteristics of RFID data pose new tasks to RFID data modeling [3,4]. RFID applications dynamically generate observations and also the data is associated with the state changes. We can consider the case of supply chain management where the inventory changes location from supplier storage facility to the end consumer. This results in a need for a data model which is able to capture the dynamic nature of RFID data as well as the relationships between different entities.

There has been interesting work on ER based temporal modeling of information system at conceptual levels. The main difference between standard ER model and temporal ER models is data changes is that the data changes very fast and has dynamic nature in the later. All entities and relationships are static or current. Entities are static but all the relationship is dynamic in a typical RFID system [3].

RFID applications are mostly associated with the timestamps when the events happen. In this paper, we propose a temporal RFID data modeling to maintain the history of events and state changes and an integrated RFID data management system.

The rest of the paper is organized as follows. Section II presents the background of the study including the RFID technology, the EPC network, and context of the supply chain. In Section III we propose temporal data modeling for RFID data. Section IV introduces essential basic operations for RFID data to monitor the states of RFID objects and to support temporal queries. Finally we conclude our study in Section V.

II. BACKGROUND

A. RFID Technology

RFID technology is classified as a wireless automatic identification and data capture technology. A RFID system consists of three layers: (i) a tag containing a chip, which is attached to or embedded in a physical object to be identified; (ii) a reader and its antennas that allow tags to be interrogated and to respond without making contact; and (iii) a computer equipped with a middleware application that manages the RFID equipment, filters data and interacts with enterprise applications [5].

RFID tags come in a large variety of designs and have many different functional characteristics. EPCglobal classifies RFID tags into 5 classes: Class 0 and 1 tags are read-only; Class 2 tags are read-write by readers; Class 3 tags come with onboard sensors and sensors can write data into tags; Class 4 tags are actually wireless sensors used for sensor networks; and Class 5 are readers. Readers may consist of a read or read/write module [6], which can initiate business processes automatically.

The RFID middleware is at the core of any RFID system. It is responsible for monitoring readers, managing, filtering, processing and aggregating all the data collected form products by readers and then

Manuscript received May 21, 2007.

Hong-Won Yun is with the Department of Information Technology, Silla University, Korea
(Email: hwyun@silla.ac.kr)
Jung-Hwa Lee is with the Department of Computer Software Engineering, Dongeui University, Korea
(Email: junghwa@deu.ac.kr)

routing the data to the dedicated information systems.

B. The EPC Network

The EPC network is a standard for RFID infrastructure [7]. It is composed of five components: (i) The EPC code starts as a 64-bit to 256-bit identifier. The EPC tags located on objects such cases and pallets. (ii) The data emerging from the EPC reader may be regarded as a stream of tuples of the form that reader, tag, and time. The EPC reader identifies any EPC tag within its interrogating field, reads the EPC tag and forwards information to the SAVANT. (iii) The SAVANT is the middleware systems located between readers and the enterprise information system. It is responsible for data filtering, aggregation and routing to the dedicated information system. The SAVANT interacts with the EPC-IS and the local ONS (Object Name Service). It was also responsible for setting up the readers. (iv) The EPC-IS is the gateway between any requester of information and the firm's information systems. This layer consists of provides higher-level services that are easier for application to use. (v) The local ONS is the authoritative directory of information sources available to describe all EPC tags used in a supply chain [7,8,9].

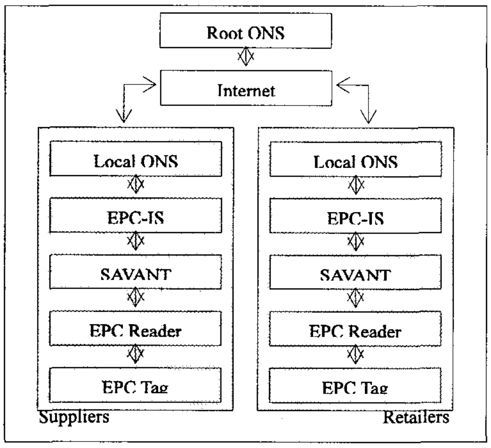


Fig. 1 EPC network architecture

C. Context of the Supply Chain

According to the Supply-Chain Operations Reference model [10] developed by the Supply Chain Council, SCM involves five core processes: plan, source, make, deliver, and return. They are conceptualized, planned, executed, and monitored at three distinct yet interrelated levels: strategic, tactical, and operational. Strategic SCM generally includes a long planning horizon and top-level executive participation.

Tactical SCM is made up of demand planning, inventory planning, and master supply planning. Operational SCM involves demand fulfillment, the procurement scheduling, production, transportation, and monitoring. A typical supply chain (see Fig. 2) consists of supplier, manufacturer, distributor, retailer, and customer [11].

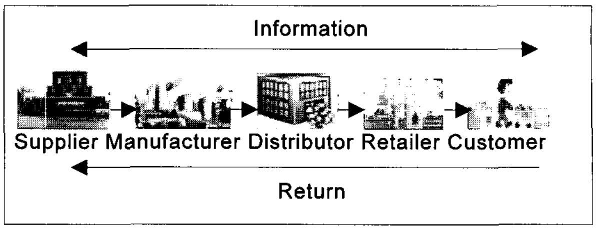


Fig. 2 Typical supply chain

A distribution center collects products from different suppliers and assembles them for delivery to a number of customer warehouses [12]. This paper focuses on a single loop supply chain in the retail industry to explore issues related to the integration of RFID technology and the EPC Network between different partners.

III. TEMPORAL DATA MODELING FOR RFID DATA

A. Fundamental Entity Types

In a distribution warehouse, four separate processes are generally identified namely shipping, put-away, picking, and receiving. All of them can benefit from RFID technology and the EPC Network. In a distribution warehouse, each product item is EPC-tagged. The tagged cases are packed onto a pallet at the supplier warehouse. And then pallets are loaded into a truck at warehouse loading zone, then the truck departs to a retail store. All pallets are unloaded from the truck at the unloading zone. All cases are unpacked from the pallets and then the cases are stocked in the store. Observations from RFID readers are made automatically and data are collected automatically in the whole process.

There can be many entities in RFID applications. We consider as fundamental entities in RFID applications. These involve objects, readers, locations, and transactions. Objects are EPC-tagged objects, such as items, cases, and pallets. The entity type object has the attributes epc, epc name, life span, and transaction time. Readers use radio-frequency signals to communicate with EPC tags and read data from tags or write data to read-write tags. It includes the attributes reader, reader name, life span, and transaction time. A location can be a geographic location or a symbolic location. Location changes come with the movement of objects and business processes. Here we assume symbolic location, such as a warehouse, and retail store. The entity type location has the attributes location identification, location name, life span, and transaction time. Transactions can be business transaction in that EPC is involved. It has the attributes transaction identification, transaction name, and life span, and transaction time.

B. Relationship Types

Relationships are generated when entities interact with each other. These include Reader_Object, Reader_Location, Object_Transaction, and Containment, as shown in Fig. 3. Most relationships in RFID applications are historical because of the temporal nature of RFID data. Reader_Object are generated when readers interact with objects. The relationship type Reader_Object has the attributes reader's epc, tag's epc, and time intervals to represent the lifespan of the state. Reader_Location are generated due to change of locations on observations. It has the attributes reader's epc, location identification, and time intervals. Object_Transaction are generated when an object participate in a transaction. The relationship type Object_Transaction involves the attributes transaction identification, object's epc, and time intervals to keep the validity of the transactions. Containment determines a hierarchical relationship among objects. For instance, cases are packed onto pallets. We use two attributes time_in and time_out or vstart and vend relate to a historical relationship. These are used to represent the lifespan of that relationship. Each relationship is associated with special time-related attributes.

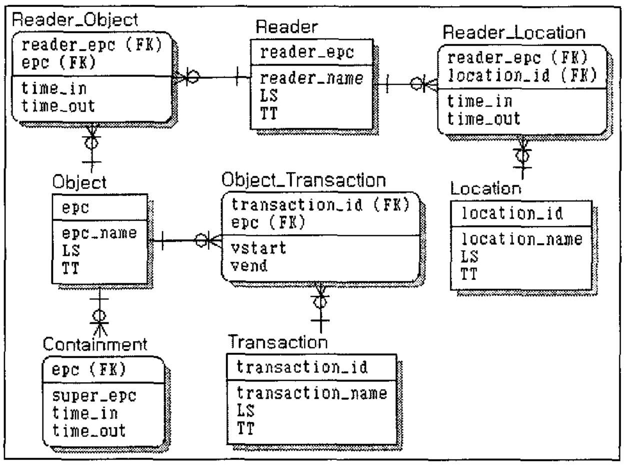


Fig. 3 Temporal ER model for RFID data

The temporal ER model for RFID data is illustrated in Fig. 3. We have following entity types based on temporal ER model: Reader, Object, Location, and Transaction. Also there are four relationship types: Reader_Object, Reader_Location, Object_Transaction, and Containment.

C. Relation Schema

This section defines the relation schema of temporal ER model for RFID data. We assume the data types *string* and *timestamp*. Our granularity for time is a millisecond. The lifespan and the transaction time of instances are specified for the entity types.

The entity types Reader, Object, Location, and Transaction are specified next. The entity type Reader has the attributes LS, TT, reader_epc, and reader_name.

Entity type *Reader* **with** (*LS*, *timestamp*), (*TT*, *timestamp*) **has**
Attribute *reader_epc* **is of type** *string*;
Attribute *reader_name* **is of type** *string*;

The entity type Object includes LS, TT, epc, and epc_name.

Entity type *Object* **with** (*LS*, *timestamp*), (*TT*, *timestamp*) **has**
Attribute *epc* **is of type** *string*;
Attribute *object_name* **is of type** *string*;

The entity type location has attributes LS, TT, location_id, and location_name.

Entity type *Location* **with** (*LS*, *timestamp*), (*TT*, *timestamp*) **has**
Attribute *location_id* **is of type** *string*;
Attribute *location_name* **is of type** *string*;

The attributes of the Transaction entity type are LS, TT, transaction_id, and transaction_name.

Entity type *Transaction* **with** (*LS*, *timestamp*), (*TT*, *timestamp*) **has**
Attribute *transaction_id* **is of type** *string*;
Attribute *transaction_name* **is of type** *string*;

The relationship types Reader_Object, Reader_Location, Object_Transaction, and Containment are as follows. The relationship type Reader_Object has the reading data generated from readers. It includes reader_epc, epc, time_in, and time_out.

Relationship type *Reader_Object* **has**
Attribute *time_in* **is of type** *timestamp*;
Attribute *time_out* **is of type** *timestamp*;
Involves *Reader*; *Object*;

The relationship type Reader_Location records the location history of a fixed reader having reader_epc, location_id, time_in, and time_out.

Relationship type *Reader_Location* **has**
Attribute *time_in* **is of type** *timestamp*;
Attribute *time_out* **is of type** *timestamp*;
Involves *Reader*; *Location*;

The relationship type Object_Transaction has attributes transaction_id, epc, vstart, and vend. It keeps information about a transaction when the transaction occurs.

Relationship type *Object_Transaction* **has**
Attribute *vstart* **is of type** *timestamp*;
Attribute *vend* **is of type** *timestamp*;
Involves *Objcet*; *Transaction*;

The relationship type Containment presents a recursive relationship that an object is held in its super type object.

Relationship type *Containment* **has**
Attribute *time_in* **is of type** *timestamp*;
Attribute *time_out* **is of type** *timestamp*;
Involves *Containment*;

Above each relationship type has a fixed time intervals.

We add key constraints to the entity types and participation constraints to the relationship type. Reader_epc is the key of Reader and epc is the key of Object. Location_id is the key of Location and transaction_id is the key of Transaction.

Reader_epc **is key of** *Reader*;
Ep **is key of** *Object*;
Location_id **is key of** *Location*;
Transaction_id **is key of** *Transaction*;
The participation constraint on Reader, Object, Location, and Transaction are (1,N).

IV. BASIC OPERATIONS FOR RFID DATA

Our temporal ER model for RFID data is highly expressive to support complex RFID queries. We propose methods to express essential RFID queries using relational algebra based on temporal ER model.

A. Event-based Query

All the relationships in a typical RFID system have dynamic nature. There are three kinds of queries as object tracking, object monitoring, and temporal query to support dynamic RFID data. We propose and summarize main queries to support the three kinds of queries as follows.

1)Current object tracking

• Find the latest location of an object with EPC value 'epc123'.

$$\Pi_{Reader\_Object.epc, Reader\_Location.location\_id} (\sigma_{Reader\_Object.epc='epc123' \wedge Reader\_Object.reader\_epc = Reader\_Location.reader\_epc \wedge max(Reader\_Location.time\_in)} (Reader\_Object \times Reader\_Location))$$

2)History object tracking

• Get the location history of an object with EPC value'epc123'.

$$\Pi_{Reader\_Object.epc, Reader\_Location.location\_id, Reader\_Location.time\_in, Reader\_Location.time\_out} (\sigma_{Reader\_Object.epc='epc123' \wedge Reader\_Object.reader\_epc = Reader\_Location.reader\_epc} (Reader\_Object \times Reader\_Location))$$

3)Object identification

• Check an object with EPC 'epc123' was sold or not.

$$\Pi_{Object\_Transaction.epc,\ Object\_Transaction.vstart,\ Object\_Transaction.vend} (\sigma_{Object\_Transaction.epc='epc123'} (Object\_Transaction))$$

The relationship type Object_Transaction's attribute *vend* value is 'now' if this product was sold, otherwise if a customer return it *vend* value has timestamp.

• Check objects with EPC 'epc123' were refund.

$$\Pi_{Object\_Transaction.epc} (\sigma_{epc='epc123' \wedge Transaction\_name='refund' \wedge Object\_Transaction.vend != 'now'} (Object\_Transaction \times Transaction))$$

4)Object snapshot query

• Find the super container of EPC 'epc123'.

$$\Pi_{Containment.super\_epc}\ \sigma_{epc='epc123'} (Object\_Transaction)$$

Object monitoring is to monitor the state of RFID objects including snapshot inquiy.

5)Object temporal query

• Get the location of an object with EPC value'epc123' at time T.

$$\Pi_{Reader\_Location.location\_id} (\sigma_{Reader\_Object.epc='epc123' \wedge Reader\_Object.reader\_epc = Reader\_Location.reader\_epc \wedge Reader\_Location.time\_in <= T \wedge Reader\_Location.time\_out <= T} (Reader\_Object \times Reader\_Location))$$

It is easy to monitor the snapshot information of any RFID objects by a timestamp. Timestamp is associated to represent the occurring timestamp of the event.

B. State-based Query

We have static entities for RFID data including Reader, Object, Location, and Transaction. RFID data can contain a lot of false readings and abnormal data in pervasive computing. The precision of objects can be precisely observed by state-based query.

1)Monitoring Physical objects

• Find the readers are not available.

$$\Pi_{Reader.reader\_epc} (\sigma_{Reader.LS != 'now'} (Reader))$$

• Get the time of a reader with reader's EPC value 'reader123' is started.

$$\Pi_{Reader.reader\_epc, Reader.TT} (\sigma_{reader.reader\_epc='reader123'} (Reader))$$

• Find the products are not available between 03.01.2007 and 03.31.2007.

$$\Pi_{Object.epc} (\sigma_{LS != 'now' \wedge Object.TT >= 03.01.2007 \wedge Object.TT <= 03.31.2007} (Object))$$

2)Monitoring Physical Locations

• Find the locations changed on 03.01.2007.

$$\Pi_{Location.location\_id} (\sigma_{location.TT= 03.01.2007} (Location))$$

• Get the locations not used now.

$$\Pi_{Location.location\_id} (\sigma_{location.LS != 'now'} (Location))$$

3)Monitoring Transaction Changes

• Find the time of a transaction with transaction's id 'tr123' is created.

$$\Pi_{Transaction.transaction\_id, Transaction.TT} (\sigma_{Transaction.TT='tr123'} (Transaction))$$

• What time was the transaction with transaction's id 'tr123' not available.

$$\Pi_{Transaction.transaction\_id, Transaction.LS} (\sigma_{Transaction.transaction\_id='tr123'} (Transaction))$$

C. Filter-based Query

RFID system with fast observation generates large volume of data. Each object has life span, an object begins his life when it is first read and ends when it is sold to customers. Sometimes RFID objects have limited life span because their life spans are over. We can classify four types of data: event-based active RFID data, event-based non-active RFID data, state-based RFID data, and state-based non-active RFID data. It is possible to achieve better query performance by filtering of non-active data.

1)Filtering event-based non-active objects

• Delete all products which move out on 03.01.2007 at location's id is 'lid123' and.

$$Reader\_Object \leftarrow Reader\_Object - \sigma_{Reader\_Location.location\_id='lid123' \wedge Reader\_Location.reader\_epc = Reader\_Object.reader\_epc \wedge Reader\_Object.time\_out=03.01.2007}(Reader\_Location \bowtie Reader\_Object)$$

• Delete all products with transaction id 'tr123' were sold on 03.01.2007 before.

$$Object\_Transaction \leftarrow Object\_Transaction - \sigma_{Object\_Transaction.transaction\_id='lid123' \wedge Object\_Transaction.vend > 03.01.2007}(Object\_Transaction)$$

2)Filtering state-based non-active objects

• Delete all transactions are not valid before 03.01.2007.

$$Transaction \leftarrow Transaction - \sigma_{Transaction.LS < 03.01.2007}(Transaction)$$

• Delete all locations have not used since 01.01.2006.

$$Transaction \leftarrow Transaction - \sigma_{Location.LS > 01.01.2006}(Location)$$

These basic operations are to support monitoring and querying over the temporal data modeling for RFID data. Especially filtering operations can obtain better query performance and efficient storage space usage.

V. CONCLUSION

RFID technology has received a great deal of attention over the last few years. It poses new challenges for RFID data management. RFID applications generate observations dynamically and the data state changes. These applications are mostly associated with the timestamps when the events happen. RFID data management systems need to have an explicit temporal data model for RFID data to support monitoring queries. We propose a temporal RFID data modeling to maintain the history of events and state changes. Also, the modeling involves basic operations for RFID data to support querying as object monitoring and temporal query for dynamic RFID data. It is possible to achieve better query performance by filtering of non-active data. The modeling in this paper focuses on a single loop supply chain in the retail industry. It can be adapted into different temporal and dynamic data applications.

REFERENCES

[1] Walmart Supplier Information: RFID Usage. http://www.walmartstores.com, 2005.

[2] L. Sparks and B. A. Wagner, "Retal exchanges: A research agenda," *Supply Chain Management*, 8(1), pp. 17-25, 2003.

[3] S. Liu, F. Wang and P. Liu, "Integrated RFID Data Modeling: An Approach for Querying Physical Objects in Pervasive Computing," CIKM'06, Nov. 2006.

[4] Y. Bai, F. Wang and P. Liu, "Efficiently Filtering RFID Data Streams," CleanDB, Sep. 2006.

[5] A. Asif and M. Mandviwalla, "Integrating the supply chain with RFID: A technical and business analysis," *Communications of the Association for Information Systems*, 15, pp393-427, 2005

[6] E. W. T. Ngai, T. C. E. Cheng, S. Au, K. -H. Lai, "Mobile commerce integrated with RFID technology in a container depot," *Decision Support System*, 43(1), pp. 62-76, 2007.

[7] EPCglobal. The EPCglobal Network, 2004. Available: http://www.epcglobalinc.org

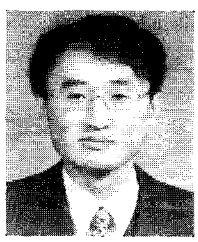
[8] B. Srivastava, "*Radio Frequency ID Technology: The Next Revolution in SCM*," Business Horizons, 47(6), pp. 60-68, 2004.

[9] Fosso Wamba et al., "Enabling Intelligent B-to-B eCommerce Supply Chain Management using RFID and the EPC Network: a Case Study in the Retail Industry," International Journal of Networking and Virtural Organizations, 3(4), pp. 450-462, 2006.

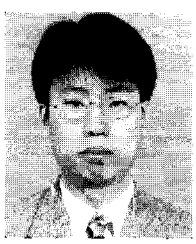
[10] *Supply-Chain Operations Reference Model: Overview of SCOR, Version 5.0.* Available: http://www.supply-chain.org/slides/SCOR5.0OverviewBooklet.pdf

[11] I. Bose and R. Pal, "Auto-ID: Managing anything, anywhere, anytime in the supply chain," *Communications of the ACM*, 48(8), pp. 100-106, Aug. 2005.

[12] V. D. Berg, J. P. and W. H. M. Zijm, "Models for Warehouse Management: Classification and Examples," *International Journal of Production Economics*, 59, pp. 519-528, 1999.



Hong-won Yun
received his B.S. and the Ph.D. degrees at the Department of Computer Science from Pusan National University, Korea, in 1986 and 1998, respectively. He is a professor at the Department of Information Technology, Silla University in Korea. His research interests include temporal database and semantic web.



Jung-hwa Lee
received his B.S. and the Ph.D. degrees at the Department of Computer Science from Pusan National University, Korea, in 1995 and 2001, respectively. He is a professor at the Department of Computer Software Engineering, Dongeui University in Korea. His research interests include database and semantic web.