Analysis and Analytical Modeling of RFID-Sensing Events for Efficient Sensor Tag Management

Md. Kafil Uddin, Wooseok Ryu, Haipeng Zhang, Bonghee Hong
Department of Computer Engineering
Pusan National University
Pusan, Republic of Korea
E-mail: {mdkafil, wsryu, jsjzhp, bhhong}@pusan.ac.kr

***Abstract*—Battery powered active Radio Frequency Identification (RFID) Tag could be used as active sensor tag integrating sensing module and using its memory to store sensing values. Thus, active sensor tag provides significant new means for carrying both RFID and sensing data. Active RFID reader is used to retrieve the id data and sensing data from the sensor tag and generate integrated RFID-Sensing events. Systems deploying sensor tag collect enormous RFID-Sensing events generated quickly from readers which eventually contain such events of several types. Although EPCglobal defines events for Electronic Product Code Information Service (EPCIS), it is not sufficient for supporting and management of integrated RFID-Sensing events. Therefore, an active sensor tag management system required to model efficiently all the events collected from the reader. In this paper, we provide analysis on active sensor tag application scenarios, related background studies and existing technologies, and all the events generated by a sensor tag through its lifecycle. Finally, we present analytical modeling of RFID-Sensing events for efficient sensor tag management.**

Keywords-sensor tag; tag memory; RFID-Sensing events.

I. INTRODUCTION

The active RFID (Radio Frequency Identification) technology is becoming more promising with the evaluation of sensor technology. The active RFID system consists of active RFID tags (i.e., transponder) and active Readers (i.e., interrogators) with active RF communication. Active tag has its own memory and on board battery which constantly powers the tag. Active tags enable a greater communication range, can be applied to metal objects, and allow easy addition of sensing modules [1]. Therefore, active RFID tag can be used as senor tag [2], enabling simultaneous id and sensing event transmission using same RF tag.

Recently, active sensor tag has found a greater interest in different application domains. A large number of such tags can be deployed in cold chain management [3], supply chain management systems and harbor logistics systems where huge amount of RFID-sensing event is generated. Consequently, sensor tag is suitable for continuously monitoring temperature-sensitive perishables, controlling product temperature and humidity, elimination of container damage due to huge inside pressure, and automatic tracking of product location. Thus, it is possible to achieve identification with sensing information using active sensor tag. However, to provide efficient system services, proper care should be taken to manage integrated events generated from various applications that use active sensor tags as a means for simultaneous id and sensing data source.

Although EPCglobal [4] defines events for Electronic Product Code Information Service (EPCIS) [5], still there is no global standard provided for RFID-Sensing event processing and its management to provide efficient system services. As a sensor tag requires storing sensing values in its memory using its own battery power, an active reader plays an important role to control and manage tag data to generate meaningful RFID-Sensing events. Moreover, active sensor tags are more vulnerable to its operations such as sensing, logging, read, write and lock operations. Furthermore, various alarm events are generated along with RFID-sensing events. Therefore, a management system requires handling and efficient modeling of all the RFID-Sensing events generated from sensor tag.

In this paper we provide analytical modeling of RFID-Sensing events and Object model for efficient sensor tag management. To do this, we analyze application scenarios of active sensor tag. We also analyze example queries relating to each scenario and all the events generated in a sensor tag lifecycle. Finally, we provide analytical modeling of RFID-sensing events and Object model for efficient sensor tag management.

The structure of this paper is as follows. Section 2 describes lists of related works. Section 3 provides a general framework where RFID-Sensing events are generated and flow of those events through the system. . In section 4, we provide analytical modeling of RFID- sensing events and Object model for efficient sensor tag management. We conclude Section 5 with avenues for further research.

II. RELATED WORKS

In this chapter we explain about the RFID event generation in EPCglobal middleware Framework [6] and EPCIS events generation in supply chain.

A. RFID Event Generation in EPCglobal Framework

The Architecture Framework [6] published by EPCglobal Inc. provides a comprehensive overview of the EPCglobal standards. The architecture framework consists of three layers: (1) Identification layer, (2) Capture layer and (3) Exchange layer; and it provides several standards for all the three layers. Each layer generates events of individual type. Tag data acquisition happens at *Identification layer* which generate physical events while *Capture layer* and *Exchange layer* generate logical events and business events respectively.

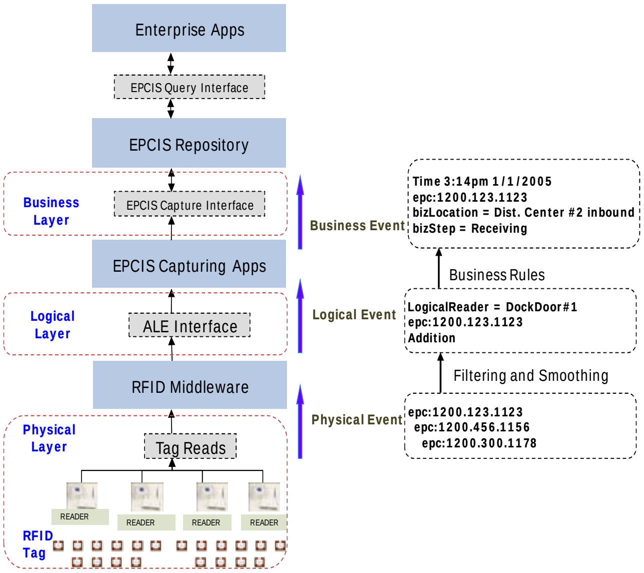


Figure 1. Generation and Flow of RFID Events in EPCglobal Framework [6]

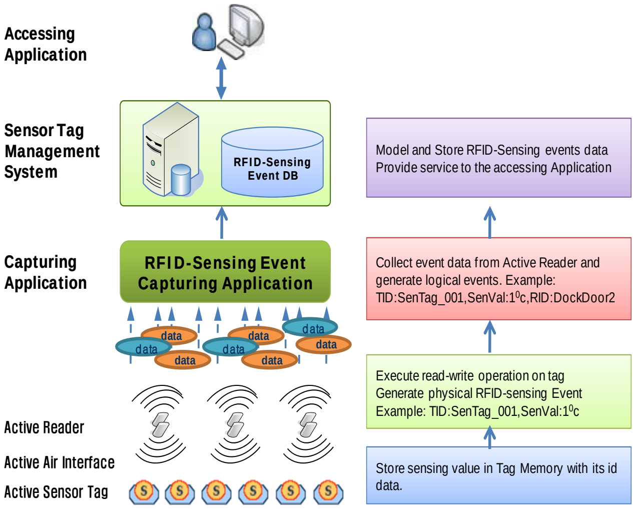
Fig.1 shows the generation and flow of RFID events through the EPCglobal Framework. Physical events are filtered at logical layer. Business rules are applied to logical events at Business layer and Business events are generated. EPCIS provides modeling of Business events to provide efficient system services to the accessing applications.

B. *EPCIS Events in Supply Chain*

In supply chain four types of EPCIS events are generated depending on the product flow and its business context. *ObjectEvent* represents an event happened to one or more entities; *AggregationEvent* represents an event that happened to one or more entities that are physically aggregated together (such as, cases are aggregated to a pallet); *QuantityEvent* represents an event concerned with a specific quantity of entities sharing a common EPC class and *TransactionEvent* represents an event in which one or more entities become associated or disassociated with one or more identified business transactions.

III. Framework for RFID-Sensing Event Flow

In this section, we present a general system framework for RFID-Sensing event generation and its flow through the system. At the bottom layer active sensor tag is used which carries sensing values in its tag memory. Active RFID reader is used to collect the id data and sensing data from the tag. Reader generates physical event and capturing application collects physical events and transforms it the logical event. Finally, logical events are stored at Sensor Tag Management system which provides system services to the accessing application. Fig.2 illustrates how RFID-Sensing events flow from active sensor tag to the management system.



Accessing Application
Sensor Tag Management System
RFID-Sensing Event DB
Capturing Application
RFID-Sensing Event Capturing Application
data
data
data
data
data
data
data
data
Active Reader
Active Air Interface
Active Sensor Tag
S
Model and Store RFID-Sensing events data
Provide service to the accessing Application
Collect event data from Active Reader and generate logical events. Example: TID:SenTag_001,SenVal:1⁰c,RID:DockDoor2
Execute read-write operation on tag
Generate physical RFID-sensing Event
Example: TID:SenTag_001,SenVal:1⁰c
Store sensing value in Tag Memory with its id data.

Figure 2. Generation and Flow of RFID-Sensing Events in Sensor Tag Management System

IV. Analysis and Analytical Modeling of RFID-Sensing Events

Sensor tag has various application domains and efficient tag management require handling various events generated to control tag and its sensing data. In this section, we present analysis of sensor tag application scenarios, management object model for sensor tag and generated RFID-Sensing events. And then we present analytical modeling for all the generated events from active sensor tag.

A. *Analysis of Application Scenarios*

In a temperature sensitive supply chain such as Cold Chain Management System (CCM) products flow from manufacturer to cold store and cold store to retailer. A cold store is similar like a warehouse where products should be carefully monitored. Here, we assume active sensor tag is attached with each pallet and cases in the pallet have passive RFID tag, shown in Fig. 3(a). The attached active sensor tag provides corresponding sensing information of the cases in a pallet, thus generates a case-pallet *Aggregation* event. Fig. 3(b) illustrates a scenario where perishable products are received, monitored and shipped from the cold store.

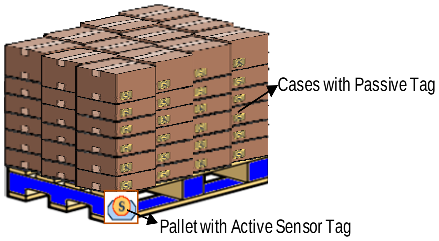


Figure 3.(a) Sensor Tag Attached with Pallet of

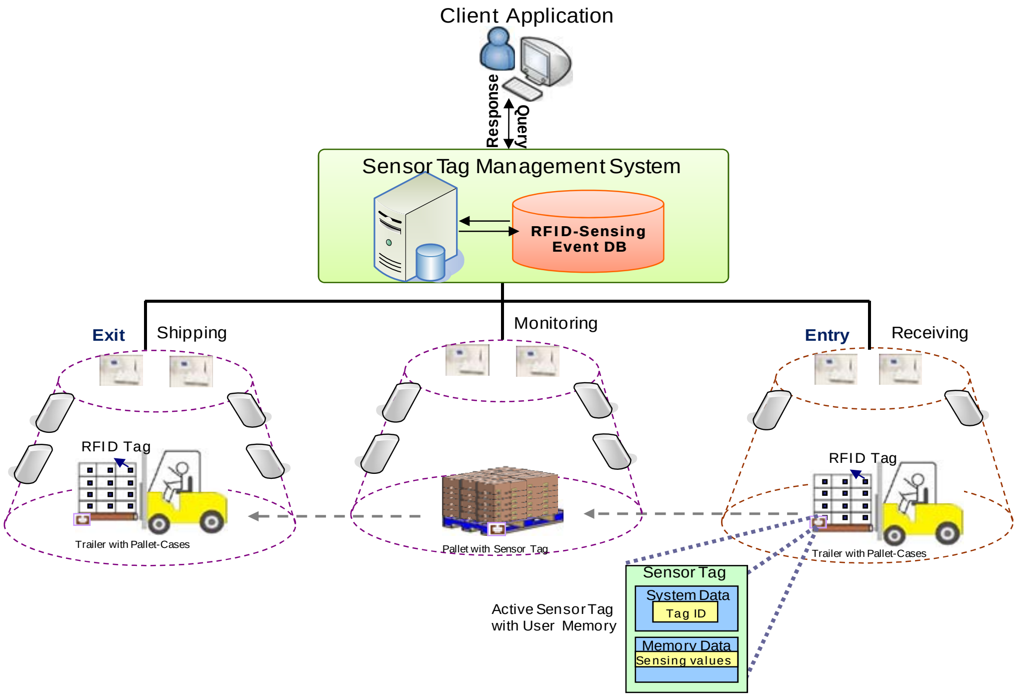


Figure 3.(b) Flow of Perishable products in Cold Store

The sensor tag management system provides query response to the user for products entering, leaving and monitored in a cold store. Table I, shows a list of user queries with corresponding generated events. An *Alarm* Event is generated if sensing value exceeds threshold limit and battery becomes low. *Quantity* event generated at Entry and Exit whereas *Sensing* event generated when any object related sensing information is queried. The sensing information of a location, such as cold store, can also be acquired from the sensor tag attached with the pallets. Here, *Zone* Event occurs when user queries for average temperature of a Warehouse. Thus five types (*Sensing*, *Alarm*, *Quantity*, *and Zone*) of events generated with case-pallet *Aggregation* event, in a cold store by using active senor tag.

TABLE I. QUERY AND EVENT TYPES IN COLD STORE (WAREHOUSE) MANAGEMENT SYSTEM

Scenario	Query	Response	Events
1. Initial Stage	No Query	No Response	No Event generated
2. Receiving at Warehouse	Report me at "each hr.", the Entry Temp. with Entry Count of all pallets in "warehouse A" from "6:00AM-6:00PM".	Entry Temperature and Entry Count	*Quantity* Event at *Entry*
3. Sensor Threshold Monitoring	Send me "alarm" if Temperature of the pallets goes above 0ºC.	Threshold Exceed alarm	*Alarm* Event
4. Battery Threshold Monitoring	Send me "alarm" if Battery of the tags goes below 1v.	Battery Low alarm	*Alarm* Event
5. Continuous Temp. Monitoring at Warehouse	3(a). Report me Temperature of pallet id:1 or pallet id:[7-10] or all pallets at each "30 min." for 5hrs from now.	monitor Sensing value "individual, range, all" tags	*Sensing Object* Event
6. Zone Temperature Monitoring	4(a). Report me Average /Minimum/Maximum Temperature of the WarehouseA at "each hr." from 6:00AM-6:00PM.	Zone Temperature (Min, Max, Avg.) Monitoring	*Zone* Event
7. Entry & Exit Temp. Monitoring	Report me at "each hr.", the Entry Temp. and Exit Temp. with Entry Count and Exit Count of pallets in "warehouse A" from "6:00AM-6:00PM".	Exit Temperature and Exit Count	*Quantity* Event at *Exit*

B. Active Sensor Tag Management Object Model

Command, Control, Notification and Alarm (CCNA) together forms the core object model for an active sensor tag management, shown in Fig. 4. In a sensor tag lifecycle, various tag read-write operations required such as setting configuration, reading tag status (battery info.).Also, management requires controlling tag states (e.g., sleeping), moods (e.g., logging), etc to store battery life.

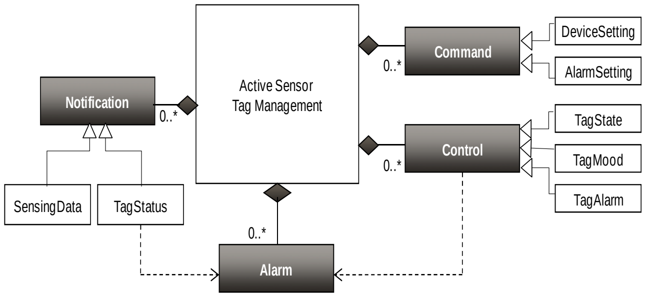


Figure 4. CCNA Object Model for Active Sensor Tag Mangement

A failure in tag operation generates alarm events by the system. There are five types of alarm event generated by the sensor tag management system. Among those first three types are related to the operation on tag failure by the manage-ment system. Tag device alarm is related to the hardware device failure and status alarm is generated with the change in tag status. The detailed description of the alarms listed the table II.

TABLE II. TYPES OF ALARM EVENTS SOR TAG MANAGEMENT SYSTEM

Tag Memory Alarm	Description
FailedTagMemoryReadAlarm	Alarms if Tag Memory read fails
FailedTagMemoryWriteAlarm	Alarms if Tag Memory Write fails
FailedTagMemoryLockAlarm	Alarms if Tag Memory Lock fails
FailedTagMemory Reset Alarm	Alarms if Tag Memory Reset fails
Tag Mood Alarm	**Description**
FailedLoggingAlarm	Alarms if Logging fails
FailedSensingAlarm	Alarms if Sensing fails
Tag State Alarm	**Description**
FailedTagActivatingAlarm	Alarms if Tag Activation fails
FailedTagSleepAlarm	Alarms if Tag Sleep fails
Tag Status Alarm	**Description**
MemoryFullAlarm	Alarms if Tag Memory becomes full
BatteryLowAlarm	Alarms if Tag battery becomes Low
Exceed Threshold Alarm	Alarms if sensing exceed the threshold
Tag Device Alarm	**Description**
NoRFCommunicationAlarm	Alarms if No RF Communication fails
DeviceMalfunctionAlarm	Alarms if Sensor Malfunctions

The design of alarm UML for handling all the alarm events generated by a sensor tag management system depicted in the following Fig. 5. The "Alarm" class provides the base alarm information of all the alarms gener-ated in the system. It provides the basic sensor tag information from which alarm is generated, the reader information and the time alarm generated. The AlarmLe-vel provides enumeration information of alarm levels such as high, low, medium etc.

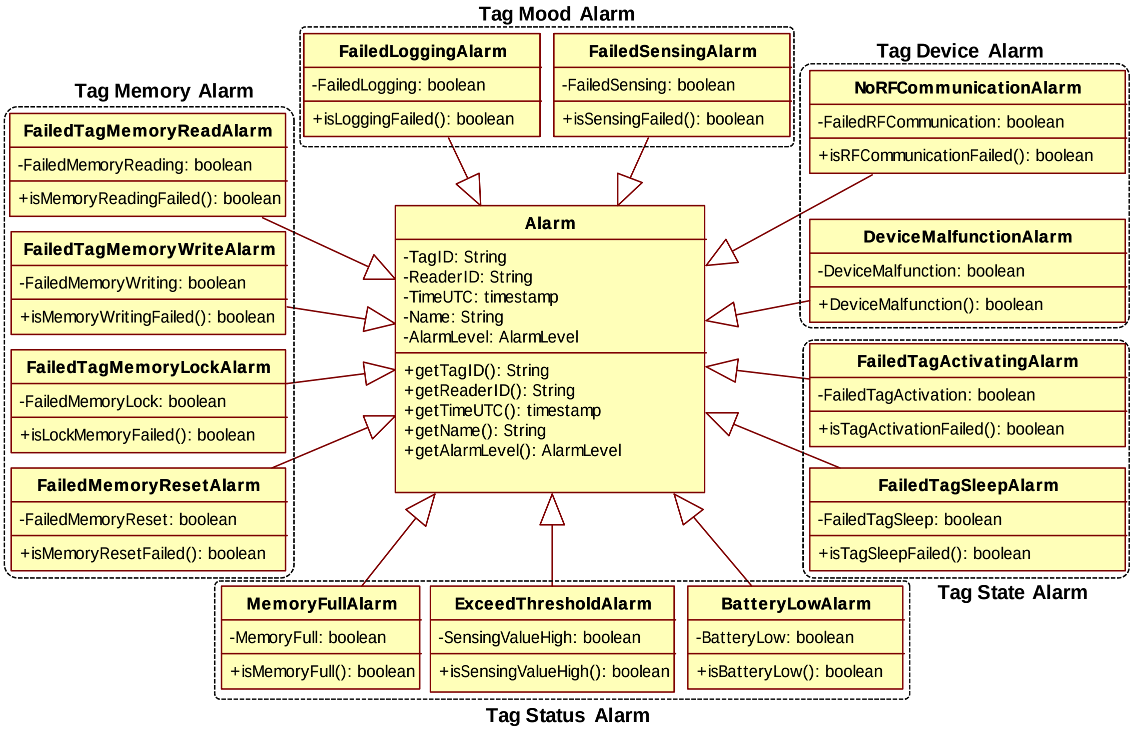


Figure 5. Alarms UML in Sensor Tag Management System

C. *Tag Management Event Model*

Five types of RFID-Sensing events generated in an active sensor tag management system. Fig. 6, depicts all the event types with data fields and data types. Event types are the names of all the events such as *SensingEvent*, *AlarmEvent*, *QuantityEvent*, *ZoneEvent*, *AggregationEvent* etc. Data field provides the field names relating to each event and data type is the type corresponding to the field. The *RFID-SensingEvent* is the generic base class for all event types generated from active sensor tag. This event contains only one data field *EventTime* with type *TimeStamp*. It provides the time when RFID-Sensing event occurred.

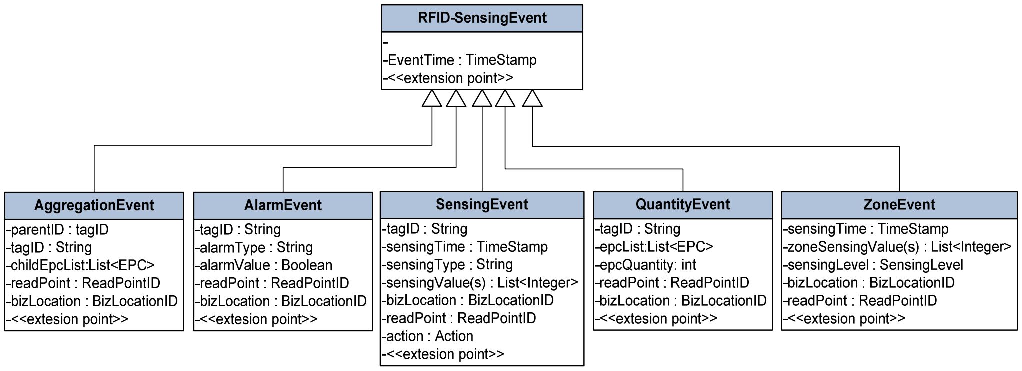
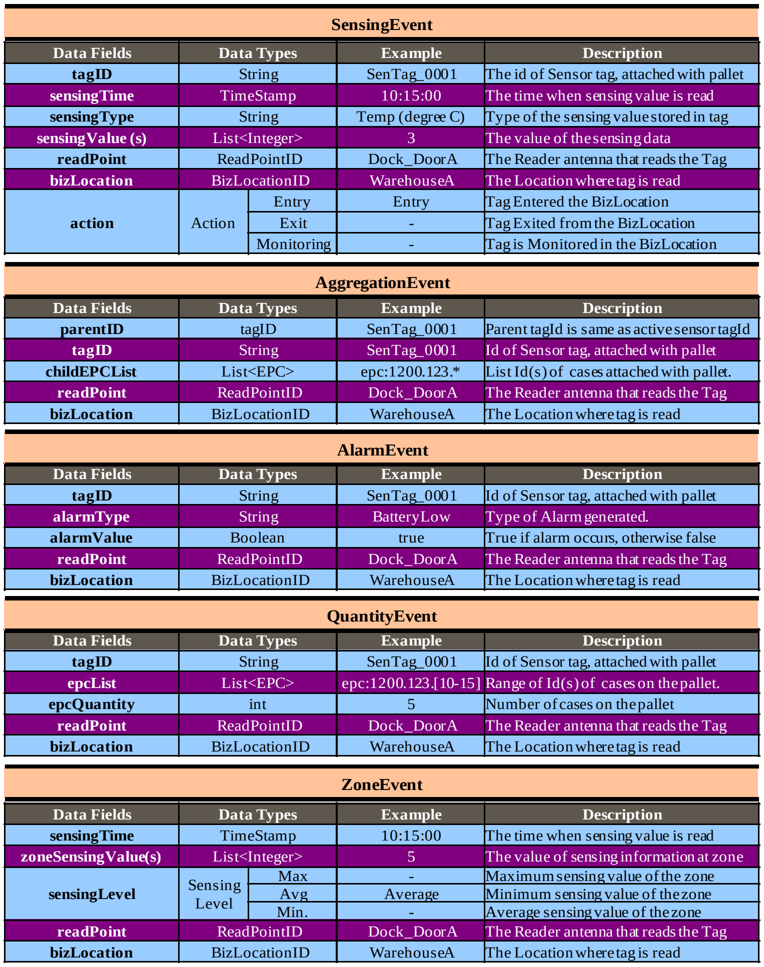


Figure 6. RFID-Sensing Events Modeling

The table III provides detailed description of all the five types of RFID-Sensing events generated by active sensor tag. We follow the standard data types provide by EPCIS standard. However, some data fields typically differ from the EPCIS data types such as *action* fields for *SensingEvent* and *sensingLevel* for *ZoneEvent.*

TABLE III. DATA FIELDS AND DATA TYPES OF RFID-SENSING EVENTS



SensingEvent

Data Fields	Data Types		Example	Description
tagID	String		SenTag_0001	The id of Sensor tag, attached with pallet
sensingTime	TimeStamp		10:15:00	The time when sensing value is read
sensingType	String		Temp (degree C)	Type of the sensing value stored in tag
sensingValue (s)	List<Integer>		3	The value of the sensing data
readPoint	ReadPointID		Dock_DoorA	The Reader antenna that reads the Tag
bizLocation	BizLocationID		WarehouseA	The Location where tag is read
action	Action	Entry	Entry	Tag Entered the BizLocation
		Exit	-	Tag Exited from the BizLocation
		Monitoring	-	Tag is Monitored in the BizLocation

AggregationEvent

Data Fields	Data Types	Example	Description
parentID	tagID	SenTag_0001	Parent tagId is same as active sensor tagId
tagID	String	SenTag_0001	Id of Sensor tag, attached with pallet
childEPCList	List<EPC>	epc:1200.123.*	List Id(s) of cases attached with pallet.
readPoint	ReadPointID	Dock_DoorA	The Reader antenna that reads the Tag
bizLocation	BizLocationID	WarehouseA	The Location where tag is read

AlarmEvent

Data Fields	Data Types	Example	Description
tagID	String	SenTag_0001	Id of Sensor tag, attached with pallet
alarmType	String	BatteryLow	Type of Alarm generated.
alarmValue	Boolean	true	True if alarm occurs, otherwise false
readPoint	ReadPointID	Dock_DoorA	The Reader antenna that reads the Tag
bizLocation	BizLocationID	WarehouseA	The Location where tag is read

QuantityEvent

Data Fields	Data Types	Example	Description
tagID	String	SenTag_0001	Id of Sensor tag, attached with pallet
epcList	List<EPC>	epc:1200.123.[10-15]	Range of Id(s) of cases on the pallet.
epcQuantity	int	5	Number of cases on the pallet
readPoint	ReadPointID	Dock_DoorA	The Reader antenna that reads the Tag
bizLocation	BizLocationID	WarehouseA	The Location where tag is read

ZoneEvent

Data Fields	Data Types		Example	Description
sensingTime	TimeStamp		10:15:00	The time when sensing value is read
zoneSensingValue(s)	List<Integer>		5	The value of sensing information at zone
sensingLevel	Sensing Level	Max	-	Maximum sensing value of the zone
		Avg	Average	Minimum sensing value of the zone
		Min.	-	Average sensing value of the zone
readPoint	ReadPointID		Dock_DoorA	The Reader antenna that reads the Tag
bizLocation	BizLocationID		WarehouseA	The Location where tag is read

1) SensingEvent: It is the core object event type of all the generated RFID-Sensing events from a sensor tag. SensingEvent generate when active sensor tag attached with any entity denoted by active tag id. Unlike EPCIS object events it contains sensing values and sensing time when the event occurred. The data field sensingType provides the type of sensing such as humidity, temperature, shock, vibration etc. The action field contains three types of actions according to the Cold Store (or, Warehouse) application. In a Warehouse, three types of actions can happen, such as product entry (Entry), product exit (Exit) and product Monitoring (Monitoring). Therefore, the three types of actions relating to SensingEvent are: Entry, Exit and Monitoring.

2) AggregationEvent: It represents an event when sensor tag attached with a product that contains items of other types with id information. In our example, we assume sensor tag attached with pallet which contains cases with passive RFID tag. The aggregated entity is denoted by the active tag id and the attached items are denoted by the child class with a list of EPCs.

3) AlarmEvent: It represents any unwanted occurrences with the sensing data, tag device, tag op-eration, tag status etc. As there are a number of alarm events generated from sen-sor tag, the alarmType field provides type information of the alarm and value is true if alarm occurs otherwise it is false. Alarm location is found from the read-Point and bizLocation data field. These are similar fields provided by EPCIS standard specification.

4) QuantityEvent: It is a "count" event that represents the count information relating to an entity where tag id used to denote identification information. Here in our application, QuantityEvent represents the number of cases attached with the pallet.

5) ZoneEvent: It represents an event where sensing value of a number of sensor tags are accumulated to measure the sensing value of a zone. It provides sensing level of a zone such as min, max and average.

V. Conclusion

Active RFID (Radio Frequency Identification) tag is becoming more promising due to its distinguishing features of long range RF communication, internal battery power and its user memory. A single RFID tag can serve double (id data and sensing data) integrating sensing module in it. Tag battery power is used to get sensing values and store in its tag memory. Thus, it is possible to achieve integrated RFID-Sensing events using active RFID tag as active sensor tag. Sophisticated applications such as Cold Chain Management Systems, harbor logistics require sensing information along with product identification information. Therefore, sensor tag can be used as a means for providing integrated RFID-Sensing events. However, it is a major system challenge to provide integrated information to the applications which requires efficient modeling of RFID-Sensing events generated from active sensor tag.

In this paper, we analyze application scenarios of active sensor tag, the generation of integrated RFID-Sensing events, flow and modification of events through the tag management framework. We also analyze existing standards and background technologies relating to RFID-Sensing events and finally we propose object model for active sensor tag management and analytical modeling for RFID-Sensing events.

Acknowledgment

This work was supported by the Grant of the Korean Ministry of Education, Science and Technology (The Regional Core Research Program/Institute of Logistics Information Technology).

References

[1] K. Finkenzeller, RFID Handbook: "Fundamentals and Applications inContactless Smart Card Identification", Second ed., John Wiley & Sons, 2003.

[2] H. Deng, "Design of Sensor-Embedded Radio Frequency Identification(SE-RFID) Systems" Proceedings of 2006 IEEE, International Conference on Mechatronics and Automation, June 25-28, 2006, Luoyang, China.

[3] http://www.sensitech.com/applications/coldchain_logistical.

[4] EPCglobal Inc., http://www.epcglobalinc.org.

[5] http://www.epcglobalinc.org/standards/epcis/epcis_1_0_1-standard-20070921.pdf.

[6] EPCglobal Inc."The EPCglobal Architecture Frame-work1.2".http://www.epcglobalinc.org/standards/architecture/architecture_1_2-framework-20070910.pdf.