ICIC Express Letters
Volume **7**, Number **3**, March **2013**

ICIC International ©2013 ISSN 1881-803X
pp. 1–ICICIC2012-894

AN EXTENSION OF RXPDL FOR RFID DEVICE MANAGEMENT EVENT HANDLING POLICIES

Haksung Kim[1] and Kwanghoon Pio Kim[2,*]

[1]Department of Computerized Taxes and Accounting
Dongnam Health University
Suwonsi Kyonggido 440-714, South Korea
amang@dongnam.ac.kr

[2]Collaboration Technology Research Laboratory
Department of Computer Science
Kyonggi University
Suwonsi Kyonggido 443-760, South Korea
*Corresponding author: kwang@kgu.ac.kr

Received June 2012; accepted August 2012

Abstract. *This paper tries to schematically extend the rXPDL* [13]*, which is an XML-based policy definition language for providing a highly abstracted interfaces to RFID applications, in order to cover the overall functionality of device management event handling policies in RFID applications. This extension is particularly aiming to be a standardized specification based upon the SSI architecture's Part-4 (RFID Application Interface) released by ISO/IEC JTC 1/ SC 31/WG 4. The extended rXPDL can eventually be an essential component of the RFID-related standardization referential framework, the SSI-RFID item management specifications of ISO/IEC standards.*
Keywords: RFID (radio frequency identification) middleware, Data management policy schema, Device management policy schema, Device interface policy schema, SSI (system software infrastructure), RFID application interfaces

1. **Introduction.** Recently, we have introduced a unified and highly abstracted RFID application interface framework [8] aiming for the RFID application developers to not only easily define RFID event-constraints, but also enhance the understandability of RFID middleware internals from the user's point of view. The framework has dubbed '*policy-driven RFID application interface framework* [8]'. In other words, we would adopt the basic concept of policy to effectively express RFID event-constraints, and to devise a series of mechanisms based upon an XML-based language defining and controlling the event-constraint policies. The XML-based language is named rXPDL [13] that stands for Policy-driven RFID Application Interface Definition Language.

As shown in Figure 1, the policy-driven RFID application interface framework [8] aims at accomplishing a higher-level abstraction in terms of implementing RFID application interfaces, and also focuses on the role of Part-4 in ISO/IEC JTC 1/SC 31/WG 4 - SSI (System Software Infrastructure) as a standardized referential framework. The essential component of the framework is rXPDL, the role of which is to provide a means to specify the interfaces between the RFID applications in the application layer and the internal blocks in the SSI platform layer, such as data management block (Part-2), device interface block (Part-3), and device management block (Part-5). The initial rXPDL [13] was designed for handling only the data management block, which is Part-2 of SSI referential framework. It is, so, needed for the initial rXPDL to be extended so as to be able to handle those events for the remainder blocks, device management block and device interface block. The scope of this paper is in extending the current rXPDL so as to particularly provide the device management functionality handling those events from the

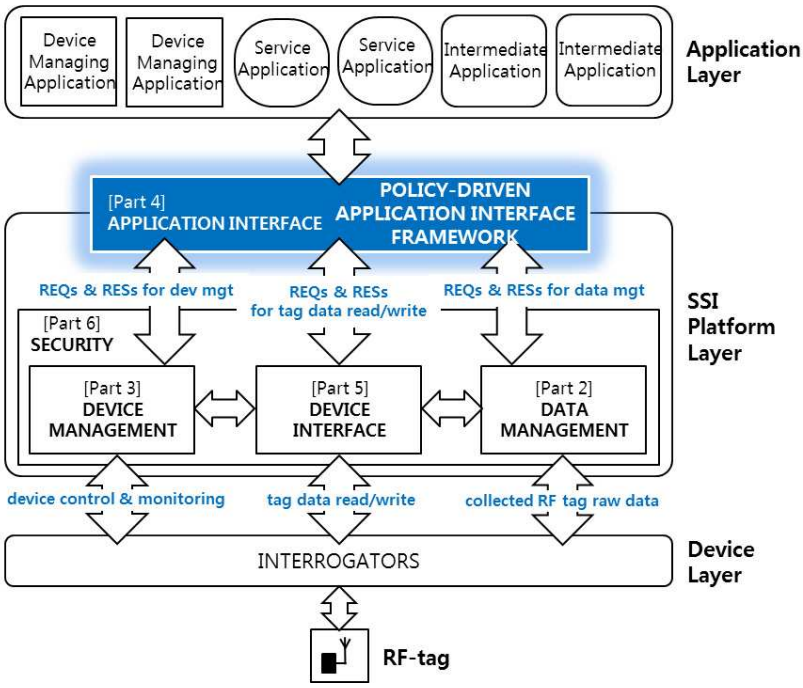

FIGURE 1. The ISO/IEC RFID middleware SSI referential framework

RFID device management block. In result, we newly devise a metamodel and schema for device management event handling policies of RFID middleware SSI internal blocks, which is extensively adopted into the current rXPDL and dubbed into rXPDL-DM, and we describe the details of the extended rXPDL-DM in this paper.

2. The Extension of rXPDL.

In this section, we try to extend the previous rXPDL [13], which was originated in supporting the data management event handling policies, so as to additionally support device management event handling policies, which is dubbed rXPDL-DM, and illustrate the details of the extended rXPDL (rXPDL-DM), such as design principles, metamodel and schema.

2.1. The current rXPDL.

In general, it is important to have a metamodel that describes all possible components and their relationships (association, inclusion and grouping) in order to build a RFID event management policy definition language. The details of the original rXPDL will not be described in this paper. Basically, the metamodel [13] for specifying a RFID data management event handling policy consists of conditions, actions and roles that are defined by the condition types of time, location, target and report, the report consumer, and the enforcement point, respectively.

Based upon the metamodel, we build an XML schema for the RFID data management event handling policy definition language. The schema [13] is hierarchically structured and syntactically represented in the Backus Naur Form. A policy is defined by a set of actions and conditions. The actions are activated by one of the types of URI (Universal Resource Identifier) – HTTP, JMS, TCP and File, while the conditions are defined by four properties – time, location, target and report. The whole details of the schema described in [13] is not described in this paper, because of the scope of the paper.

2.2. The extended rXPDL: rXPDL-DM.

A RFID device management event handling policy is defined by the rXPDL-DM metamodel given in Figure 2. The metamodel is configured by three types of policy-condition, policy-action, and role. Particularly, the policy-condition type specifies time, location, target, report, and event constraints,

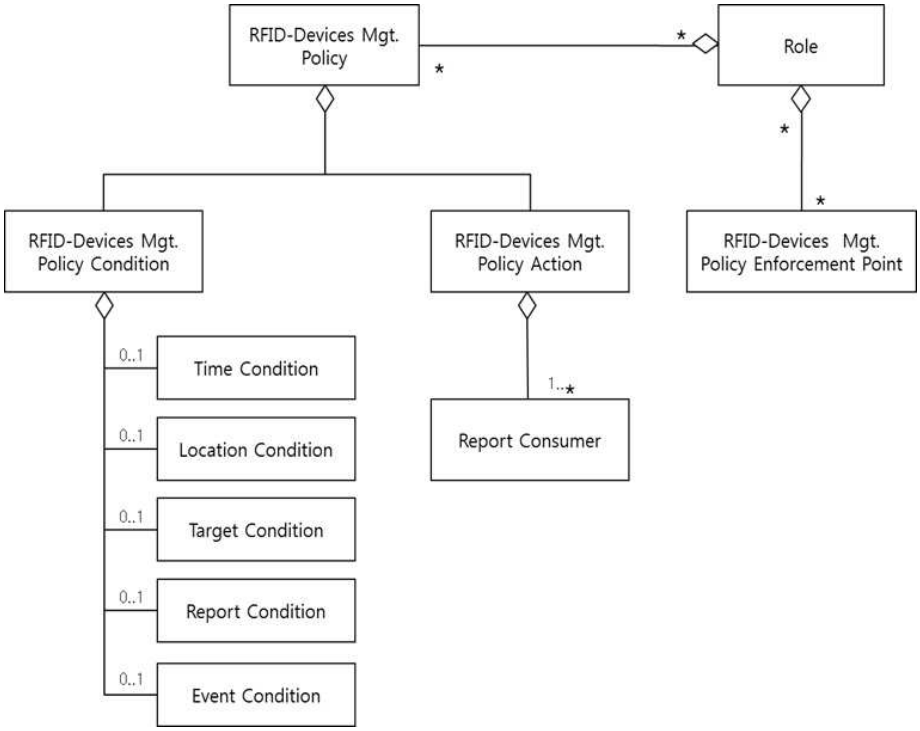

FIGURE 2. rXPDL-DM metamodel for device management policies

and also the policy-action type can be defined by one or more report consumers. Additionally, the policy's security is defined through a role that controls the accesses to the corresponding policy enforcement points connecting to the underlying RFID devices.

2.3. **Metamodel of the rXPDL-DM.** The detailed explanations of the policy-condition types can be simply described as follows:

- Time condition type is used to set a time range to collect the status information coming from multiple devices and to also specify time intervals to receive the status information within the set time range.
- Location condition type is used to include logical locations to be managing multiple RFID devices, and to specifically exclude any logical locations to remove the unneeded RFID devices. Note that the location condition can be specified by referring users to the values of logical locations representing the physical arrangements of the underlying RFID devices.
- Target condition type is used to include some RFID tags to be managed by a specific RFID device, and to exclude any RFID tags from the managed tags by the specific RFID device.
- Report condition type is used to specify a refining style of the collected RFID status information.
- Event condition type is used to specify a status request condition for managing multiple RFID devices, and to remove any status conditions to be unneeded anymore in managing the RFID devices.

Based upon the metamodel of RFID device management event handling policies, we develop the rXPDL-DM's schema for syntactically defining a RFID device management event handling policy, which stands for XML-based RFID Policy Definition Language for Device Management. The following BNF-styled statement represents the highest-level of the rXPDL-DM's schema, and the statement can be easily transformed into an XML schema. The detailed description of the elements of a DM-policy statement is summarized in Table 1.

Table 1. Description of the elements in the DM-policy statement

Element	Option	Description
DM-POLICY-NAME	Requested	Required Element to be uniquely used for identifying each of RFID device management policies.
DM-POLICY-ROLES	Optional	Default Element to be managed by the system's role when the role is not specified.
DESCRIPTION	Optional	Additional explanation about the corresponding RFID device management policy.
DM-POLICY-CONDITIONS-STATEMENT	Optional	Zero or more DM-policy-conditions describing a specific situation through time, location, target, report, and event condition units.
DM-POLICY-ATIONS-STATEMENT	Requested	Required Element to be provided as setvices, and it can be a compound DM-policy-action containing multiple units of DM-policy-actions.

```
0:  < DM-POLICY-STATEMENT >
1:     ::= DMPolicy {
2:        < DM-POLICY-NAME >
3:        < DESCRIPTION >
4:        < DM-POLICY-ROLES >
5:        < DM-POLICY-CONDITIONS-STATEMENT >
6:        < DM-POLICY-ACTIONS-STATEMENT >
7:      }
8:  < DM-POLICY-CONDITIONS-STATEMENT >
9:     ::= DMPolicyConditions {
10:        < CONDITION-OP-TYPE >
11:        < DM-POLICY-CONDITIONS >
12:      }
13:  < DM-POLICY-ACTIONS-STATEMENT >
14:     ::= DMPolicyActions {
15:        < ACTION-OP-TYPE >
16:        < DM-POLICY-ACTIONS >
17:      }
```

```
0:  < DM-POLICY-CONDITIONS >
1:     ::= <> | < DM-POLICY-CONDITION-STATEMENT > |
2:       < DM-POLICY-CONDITIONS >< DM-POLICY
                    -CONDITION-STATEMENT >
3:  < DM-POLICY-CONDITION-STATEMENT >
4:     ::= DMPolicyCondition {
5:        < DM-POLICY-CONDITION-NAME >
6:        < DESCRIPTION >
7:        < DM-TIME-CONDITION-STATEMENT >
8:        < DM-LOCATION-CONDITION-STATEMENT >
9:        < DM-TARGET-CONDITION-STATEMENT >
10:        < DM-REPORT-CONDITION-STATEMENT >
11:        < DM-EVENT-CONDITION-STATEMENT >
12:      }
```

```
0:  < DM-POLICY-ACTIONS >
1:     ::= <> | < DM-POLICY-ACTION-STATEMENT > |
```

```
2:       < DM-POLICY-ACTIONS >< DM-POLICY
                    -ACTION-STATEMENT >
3:  < DM-POLICY-ACTION-STATEMENT >
4:     ::= DMPolicyAction {
5:        < DM-POLICY-ACTION-NAME >
6:        < DM-REPORT-CONSUMER >
7:        < DESCRIPTION >
8:     }
```

2.4. **APIs for RFID device management.** In this section, we simply introduce a set of APIs for supporting the RFID device management policies. Due to the page limitation, we would not describe the details of the APIs.

```
1:  GetSources():  List<String>
2:  GetAllPropertyMetadata(SourceName:  String):  List<String>
3:  GetPropertyMetadata
    (SourceName:  Stirng, PropertyIdentifiers:  String) :  List<String>
4:  GetAllPropertyValues   (SourceName:  String):  List<String>
5:  GetPropertyValue
    (SourceName:  String, PropertyIdentifiers:  String):  List<String>
6:  SetPropertyProfile
    (SourceName:  String, PropertyProfile:  List<String>):  void
7:  GetPropertyValuesByGroup
    (SourceName:  Stirng, GroupURI: String):  List<String>
8:  Reboot() :  void
9:  ResetToFactorySettings() :  void
10:  ResetToFactorySettingsExceptNetwork() :  void
```

3. **Related Works.** So far, there exist several researches related with this topic for easily specifying RFID event-constraints by furnishing with a high-level abstraction, which is generally called the rule-based RFID service management frameworks and systems. That is, Wang et al. [7] proposed the rule condition and method of action definition for RFID service integrations according to the specific events obtained from the context elements and event refinement declarations with respect to time, location and target. His approach also supported the negative condition syntax related to the tense constraints. However, the declaration part of the event refinement method takes into account just time, location and target, and did not support the method of report conditions that are considered in this paper. Therefore, his approach may be unsuitable for the service domains possibly generating a huge amount of events. Xu et al. [6] suggested the operation method. The method detects from several RFID readers by applying the Manchester coding algorithm after allocating a collection pattern about a large amount of low events, and then it makes running through query retrievals for the corresponding business rules that are registered previously after analyzing some derivations between patterns of relevant events. However, the proposed method has some disadvantages in terms of that it needs a certain amount of high-capacity storage in order to analyzing patterns in a large amount of low events.

Additionally, Moon et al. [11] proposed the CEL (Contextual Event Language) description language to define RFID events, which is satisfied with the registered event cycle specifications such as the event refinement specifications and rules. However, the proposed method may have some limitations in terms of that the users ought to completely understand the structures of the event cycle specifications in order to define the rule conditions. Finally, Kim et al. [12] proposed the description method that can define not only the condition syntaxes of business rules as BESpec (Business Event Specification), but also the corresponding event refinement specifications according to the previously registered event cycle specifications. And it can integrate the performance results of BESpec into a composite web service defined in BEReport (Business Event Report). However,

the users should understand the structures of ECSpecs and ECReports in order to define the BESpec.

Recently, Ahn et al. [13] proposed an XML-based policy definition language for policy-driven RFID event management agents, which was named as rXPDL, to provide a reasonable way to easily specify RFID event-constraints with a high-level abstraction. This paper is an extended version of rXPDL to cope with the RFID device management functionality.

4. **Conclusions.** So far, this paper has presented an extension of the highly abstracted RFID event description language, which is called rXPDL, to adopt the RFID device management event handling functionality into the SSI platform by defining its meta-model and XML schema. Particularly, in this paper we pointed out the limitations of the current RFID event management processing frameworks and standardized specifications, and proposed a feasible solution for resolving the limitations through the concept of policy and its related details. Conclusively, the extended rXPDL is able to also convey more convenient, intelligent and broad-ranged RFID device management services to the RFID application developers and managers. It also is strongly believed that the extended language and its runtime framework should be an impeccable solution for realizing the useful, effective, flexible, and extensible RFID events management systems.

Acknowledgement. This research was supported by the Gyeonggi Regional Research Center, Contents Convergence Software Research Center, at Kyonggi University, funded by the Gyeonggi Province, Republic of Korea.

REFERENCES

[1] EPCglobal. Inc, *EPC Information Services (EPCIS) Version 1.0.1 Specification*, 2007.
[2] EPCglobal. Inc, *The Application-Level Events (ALE) Specification, Version 1.1 – Part 1: Core Specification*, 2008.
[3] EPCglobal. Inc, *The Application-Level Events (ALE) Specification, Version 1.1 – Part 2: XML Bindings*, 2008.
[4] EPCglobal. Inc, *EPCglobal Object Name Service (ONS) 1.0.1, Ratified Standard Specification with Approved*, 2008.
[5] S. Wamba, L. A. Lefebvre and E. Lefebvre, Enabling intelligent B-to-B e-Commerce supply chain management using RFID and the EPC network: A case study in the retail industry, *ACM International Conference Proceeding Series*, vol.156, 2006.
[6] J. Xu, W. Cheng, W. Liu and W. Xu, XML based RFID event management framework, *Proc. of IEEE Region the 10th Conference TENCON 2006*, vol.1, pp.1-4, 2006.
[7] F. Wang, S. Liu, R. Liu and Y. Bai, Bridging physical and virtual worlds: Complex event processing for RFID data streams, *Proc. of Advances in Database Technology – EDBT 2006, Lecture Notes in Computer Science*, vol.3896, pp.588-607, 2006.
[8] H. Ahn et. al, A policy-driven RFID event management framework, *Proc. of ICUT2009, IEEE Computer Society*, 2009.
[9] ISO/IEC JTC 1/SC 31/WG 4 24791-2, *System Software Infrastructure: Data Management Specification*, 2008.
[10] ISO/IEC JTC 1/SC 31/WG 4 24791-4, *System Software Infrastructure: Application Interface Requirements*, 2008.
[11] M. Moon et al., Contextual events framework in RFID system, *Proc. of the 3th International Conference on Information Technology: New Generations*, pp.586-587, 2006.
[12] S. Kim et al., RFID business aware framework for business process in the EPC network, *Proc. of the 5th International Conference on Software Engineering Research, Management and Applications*, pp.468-475, 2007.
[13] H. Ahn et al., **rXPDL**: An XML-based policy definition language for policy-driven RFID event management agents, *IEEE Proc. of ICACT2011*, 2011.