CHAPTER 7
DESIGN AND IMPLEMENTATION OF AN APPLICATION LAYER ORIENTED ARCHITECTURE FOR RFID SYSTEMS

7.1 INTRODUCTION

This part of the work proposes an Application layer oriented architecture and its implementation accepting one of the techniques proposed in Chapter 4 of the thesis namely Using Fuzzy methodology to control the uncertainty in the RFID data. In RFID systems, primarily four layers are in actual function if compared with the standard OSI model, i.e. Physical layer, Data Link Layer, Presentation and Application Layer. The proposed architecture work focuses on the modeling of application layer of RFID systems.

7.2 PROPOSED APPLICATION LAYER ORIENTED ARCHITECTURE FOR RFID

The proposed architecture is depicted in Figure 7.1. This architecture is proposed by keeping in view the present RFID systems and their associated issues. Some of such issues are highlighted and worked upon throughout this thesis. This architecture reflects not only the real-time scenario of the present (2011) RFID systems but also bears the capability of handling various kinds of issues associated with such systems.

The upcoming sections in sequence come up with the detailed description and block diagram of the proposed application layer oriented architecture for RFID systems.
7.2.1 DESCRIPTION

This architecture assumes inputs from an RFID emulator in the RFID scenario to appreciate the reduction of development cost and time. A finite state machine in the RFID Reader Interface is depicting the functionality of the RFID Reader Interface. This stage is responsible for event generation and invocation of the application layer of RFID middleware initially.

The output of this stage is Raw RFID Events which are further being passed on to the Sensor Abstraction Layer.

The State machine depicted in the RFID Reader Interface allows each RFID tagged device to be searched for and selected close to the antenna RFID reader also allows three main actions namely, read, write and get ID. A polling mechanism is also being used in which a user may be provided with an option to select how many times an RFID tag in the sensing scenario may be searched.

The proposed machine searches for a tag, if the tag is found a process is then invoked to recover the tag. The tag then would respond with its unique identification number, and is then capable of performing the specified requested actions. The events so generated by this machine may be then fed as an input to the sensor abstraction layer or directly to the data processing subsystem of the RFID Middleware.

Now, since the RFID scanning is in real time and always on, and the RFID observations have implicit meanings along with the temporal and streaming property of RFID data which have to be processed on the fly [131], a real time data processing layer framework is designed and accommodated in the proposed architecture.

This helps to automate the transformation of physical RFID observations into the virtual counterparts in the virtual world linked to business applications and even green IT applications [8,40,155,165,166,167].
This data processing layer has its own storage capabilities and is incorporated with embedded intelligent queries fires by an expert user. These rules are embedded into application logic to be inputted into complex event processing engine for RFID systems. Further, since RFID is real-time data which always needs attention, listeners are designed and embedded to take care of the same.

Apart from that, the discrete events when looked one by one might be meaningless in case of RFID systems, event streams which is an infinite set of events are considered over a sliding window and further correlated, are embedded in the proposed architecture.

Also, in continuation the minimal response time reaction is captious for impressive action plus cut-throat benefit in RFID systems. The complex event processing engine in the proposed architecture produces intelligent output to the Application interface here which may be directly consumed by the business applications for greater purposes.

7.2.2 BLOCK DIAGRAM

Figure 7.1 presents the block diagram of the proposed application layer oriented architecture for RFID systems.
Figure 7.1: An Application Layer Oriented Novel Architecture for RFID
7.3 ADVANTAGES AND LIMITATIONS OF THE PROPOSED ARCHITECTURE

The advantages associated with the proposed architecture are many fold. The proposed architecture may act as a base tool for generating next generation business process management and automation of sensor network applications by companies in the implementation phase of this technology. Such applications may then also be offered as cloud services over the internet. Several optimizations have been incorporated while implementing this architecture which will be beneficial in optimizing the RFID system performance without actual upgradation of hardware hence reducing the time and cost associated. One of very important advantages associated with this architecture is that the RFID event pattern matching and event visibility can be very much controlled by the timers and via operations. This architecture also provides a direct insight into the real-time working of RFID systems bearing the capability of handling the performance issues associated with the present (2011) RFID systems. The proposed architecture can also handle the RFID issue which is proposed in Chapter 6 with little modifications at the Sensor Abstraction Layer.

The limitations include support for cloud services which may be offered at application layer.

7.4 IMPLEMENTATION OF THE PROPOSED ARCHITECTURE

The implementation phase of the proposed architecture has been divided into two sections with one bearing the system and software requirements and another composed of the implementation details and the result snapshots.

7.4.1 SYSTEM REQUIREMENTS AND SOFTWARE REQUIREMENTS

The implementation of the architecture has been carried out on the following system specifications and software requirements:
The software which are used to implement the proposed application layer oriented architecture for RFID are Open Source.

7.4.2 IMPLEMENTATION DETAILS

This work focuses on integrating, detailing and implementation of the RFID application layer oriented architecture proposed above. Here, in this work a virtual shopping mall namely, All-In-One Mall (AIOM) has been created. This mall has two main entrances namely AIOM Entrance 1 and AIOM Entrance 2 from where the RFID tagged items are shipped in for further processing. Also, in association with the AIOM, a sensing scenario configuration with default properties has been created with two Alien 9800 readers with 127.0.0.1:20000 and 127.0.0.1:20001 ports and associated commands and SGTIN EPC GEN2 RFID tags with unique Tag IDs are taken into consideration. These readers are successfully responding on poll commands and their respective sessions are starting in order to establish a TCP/IP connection with the Entrance2 reader.

An Intelligent RFID application has been designed which takes input in the form of raw RFID events from the sensor abstraction layer and forwards it to the data processing unit of the
proposed architecture for further processing. At this stage intelligence is embedded to accommodate expert user queries. This phase is drawn from chapter 5 accelerating technique namely “Using fuzzy methodology to control the amount of uncertainty in RFID data”. Further, the mined rules are finally forwarded to the RFID complex event processing engine for subscribing, streaming and listening of the real time RFID events. Tag read duplicity elimination has been incorporated. Apart from that, even if it so happens that the RFID tag arrived is not EPC GEN2 RFID tags, then also the application would recognize those tags. The event patterns are incorporated to be repeated for every tag that appears at any of the entrances of AIOM. RFID event streaming has been taken care of by the listeners designed for the events at both the AIOM Entrance events.

Arrivals and departures are managed by real-time updating of the windows. The design of services is attained that the controller uses. Then Injection of the RFID tag location services is performed into the controller. Distinct alerts are designed and invoked to display alerts in various needful situations like theft etc. These alerts are then passed over a Java Messaging Service for a Web Application and dynamic web page creation. Two plugins were designed to provide the intelligent RFID service to the user namely com.intelli.rfid.tracking and com.intelli.rfid.tracking.war.

The implementation of the proposed architecture holds assumptions like while the ReadCycle of an RFID system is normally constituted of more than one individual event but here a technique has been used which is capable of returning only the RFID tags (in each ReadCycle). Also, it has been assumed in the considered AIOM scenario that if a tag is not visible for some specified duration of time, it might have moved from AIOM Entrance1 to AIOM Entrance 2.

7.4.3 OPTIMIZATIONS PERFORMED WHILE IMPLEMENTATION

A number of optimizations have been incorporated while implementing the proposed architecture. Firstly, the overall logging overhead has been reduced by setting the logging level to “WARN”. Since the RFID data pose the characteristics of streaming data, stream-level data
filtering has been used instead of using the post-data-window filtering. Apart from that, the AIOM RFID events have been selected instead of working out on individual fields to boost the performance of the system.

### 7.4.4 THE IMPLEMENTATION RESULTS

Figure 7.2 illustrates the process of successfully initializing the bundles and making them ready for execution.

![Figure 7.2: Snapshot of Running Designed Configurations for AIOM](image)

106
Figure 7.3 of this chapter depicts the process of successful launch of the application framework consisting up of the designed bundles namely com.intelli.rifidi.tracking and com.intelli.rfid.tracking.war which are highlighted in the figure and numbered 4 and 5 respectively.

Figure 7.4 presents the snapshot of successful TCP/IP connection establishment between the two Alien 9800 Readers namely AIOMReader1 and AIOMReader2 fixed at AIOMEntrance1 and AIOM Entrance2 respectively.

It also shows that the session establishment with both the Alien 9800 RFID Readers has taken place.

![Figure 7.3: Snapshot of AIOM RFID Application Framework Launched](image)
Figure 7.4: Snapshot of AIOM Alien 9800 Reader Sessions

Figure 7.5: Snapshot of Tags Reading performed by AIOM Alien 9800 Readers
Figure 7.5 demonstrates the live reading of incoming RFID tags by both the RFID readers placed at AIOM Entrance1 and AIOM Entrance2 tags.

Figure 7.6 presents the creation of a dynamic web page of AIOM Demo on the fly which shows the unique RFID tags represented by the RFID tag identification numbers which as and when are appearing on the respective AIOM Entrances.

![AIOM Demo](Image)

**Figure 7.6:** Snapshot AIOM Demo raising High Alerts
According to the configured scenario and embedded intelligence the alerts are displayed for example if the AIOM package passed AIOM entrance 2 but never reaches AIOM Entrance 2, an alert is raised and a false positive read which otherwise would have been popped up has been eliminated. Hence, this implementation clearly depicts that the uncertainty is clearly eliminated.

The existing work [162,165] that was found nearer to the proposed work was missing two very important things in context of RFID business applications:

Firstly, there was no expert user feedback provision (Fuzzy Controller) in terms of requirement analysis to update the real time working and embedding intelligence in RFID applications.

Secondly, no use of optimization techniques like reducing the overall logging overhead or use of RFID events instead of working on individual fields has been done. These things are incorporated in the proposed architecture.

7.5 CONCLUSIONS

At the application layer of RFID systems which itself included the RFID Interface Layer, Sensor Abstraction Layer, Data Processing layer and the application interface layer, this chapter proposed a magnified view of the RFID proposed architecture (including the embedding of the architectural requirements, optimization and intelligence in the current RFID systems). This chapter also presented strong implementation results like uncertainty and duplication elimination along with the use of this additional information in the present (2011) RFID applications. Also, the implementation of the proposed architecture as well supports the existence and further scope of implementations in RFID systems by also resolving some of the performance issues and incorporating more optimizations at software level.