# Chapter 19
# An Integrated Development Environment for RFID Applications

**Nikos Kefalakis**
*Athens Information Technology, Greece*

**John Soldatos**
*Athens Information Technology, Greece*

## ABSTRACT

*In recent years we have witnessed a proliferation of RFID (Radio Frequency Identification) middleware systems and projects (including several open source projects), which are extensively used to support the emerging wave of RFID applications. Some of the RFID middleware projects come with simple tools, which facilitate the application development, configuration, and deployment processes. However, these tools tend to be fragmented since they address only part of an RFID system (such as the filtering of tag streams and/or the generation of business events). In this chapter, the authors introduce an Integrated Development Environment (IDE) for RFID applications, which addresses multiple parts of an RFID application, while at the same time supporting the full application development lifecycle (i.e. design, development, deployment, and testing of RFID applications). The introduced IDE comprises a wide range of tools, which have been implemented as modular plug-ins to an Eclipse-based environment. The various tools enable application development, deployment, testing, and configurations over the middleware infrastructure established by the AspireRFID (AspireRFID Consortium, 2013), and their evaluation has proven that they can significantly ease RFID application development.*

## 1. INTRODUCTION

During recent years we are witnessing a proliferation of successful RFID deployments in areas such as manufacturing, logistics, trade and industry. Several of these deployments deal with numerous tags/objects, RFID readers, reading cycles and RFID generated events, which route to a wide range of enterprise

DOI: 10.4018/978-1-5225-3422-8.ch019

applications such as ERP (Enterprise Resource Planning), MRP (Manufacturing Resource Planning) and WMS (Warehouse Management Systems) (Kefalakis, 2009). RFID middleware is the cornerstone of most of the above non-trivial deployments of RFID technology. RFID middleware undertakes crucial tasks in the scope of RFID applications, which typically include (Dimitropoulos, 2010):

- Collecting RFID data from a variety of heterogeneous physical readers, through reading the tagged items. At this level middleware implementations insulate higher layers from knowing what reader /models have been chosen. Moreover, they achieve virtualization of tags, which allows RFID applications to support different tag formats.
- Filtering the RFID sensor streams according to application needs and accordingly emits events pertaining to the application at hand. At this level middleware implementations insulate the higher layers from the physical design choices on how tags are sensed and accumulated and how the time boundaries of events are triggered.
- Mapping the filtered readings to business semantics as required by the target applications and business processes. The respective business semantics are consolidated into business events, which are routed (by the RFID middleware) to one or more enterprise applications that use them. At this level middleware implementations insulate enterprise applications from understanding the details of how individual steps are carried out in a business process.

Note that most of these middleware tasks are defined in the scope of the EPC Global Architecture (EPCGlobal Architecture Review Committee, 2013), which is supported by a wide range of relevant standards. The importance of RFID middleware, has given rise to the development of several RFID middleware frameworks, which are nowadays providing functionality for RFID data collection, filtering, event generation, as well as translation of tag streams into business semantics. These frameworks have been developed as part of both research initiatives (Prabhu, 2006) and vendor products. Furthermore, several research initiatives have produced numerous open-source RFID frameworks, such as Mobitec (Mobitec, 2013) and the Fosstrak project (Floerkemeier, 2007), which provide royalty-free implementations of RFID middleware stacks. In addition to providing middleware libraries for RFID applications, most of these projects provide simple tools for configuring and managing the underlying RFID infrastructures. These tools facilitate application development and monitoring, yet they do not obviate the (still needed) tedious low-level tasks.

Apart from RFID middleware projects, RFID initiatives have also emerged, focusing on application development tools (e.g., LogicAlloy (2013), Rifidi (Palazzi, 2009) and its successor projects). In several cases, these initiatives also offer RFID middleware libraries (e.g., the LogicAlloy (2013) Server), while in other cases they are mostly focused in visual RFID application development. Despite the emergence of these efforts, there are no integrated environments for RFID deployment. As a matter of fact, most of the above tools are focused on parts of the RFID middleware development effort (e.g., the configuration of filtering mechanisms, the management of metadata and the management of reader devices) rather than on a holistic approach to RFID solution deployment. As a result, most of RFID middleware solutions are implemented on a per-case basis, rather than based on a more convenient and cost-effective integrated paradigm.