LIT Middleware: Design and Implementation of RFID Middleware based on the EPC Network Architecture

Ashad Kabir, Bonghee Hong, Wooseok Ryu, Sungwoo Ahn

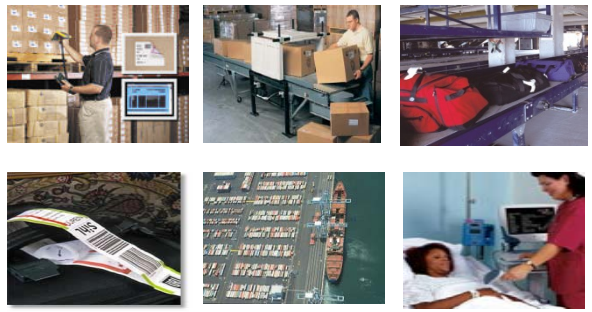
Dept. of Computer Engineering

Pusan National University, Republic of Korea

Introduction

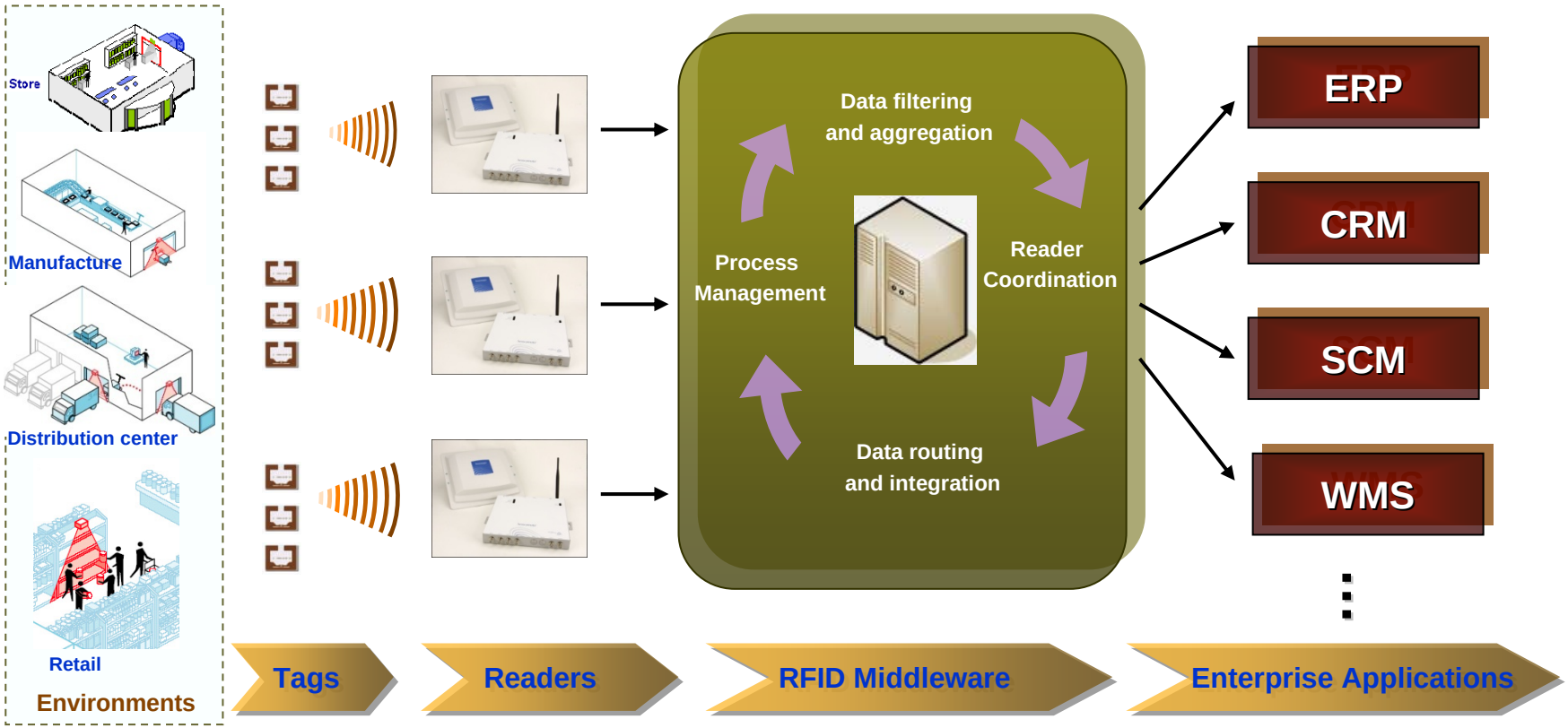
- **RFID Middleware needed for**
  - Track & Trace item automatically throughout the supply chain.
  - Safe & Secure supply chain.
  - Fast & Accurate data collection.
  - Business process automation.

- **Application Domains**
  - Supply Chain Management (SCM) in transportation
  - e-Government
  - e-Health
  - Waste management
  - Documents identification & tracking

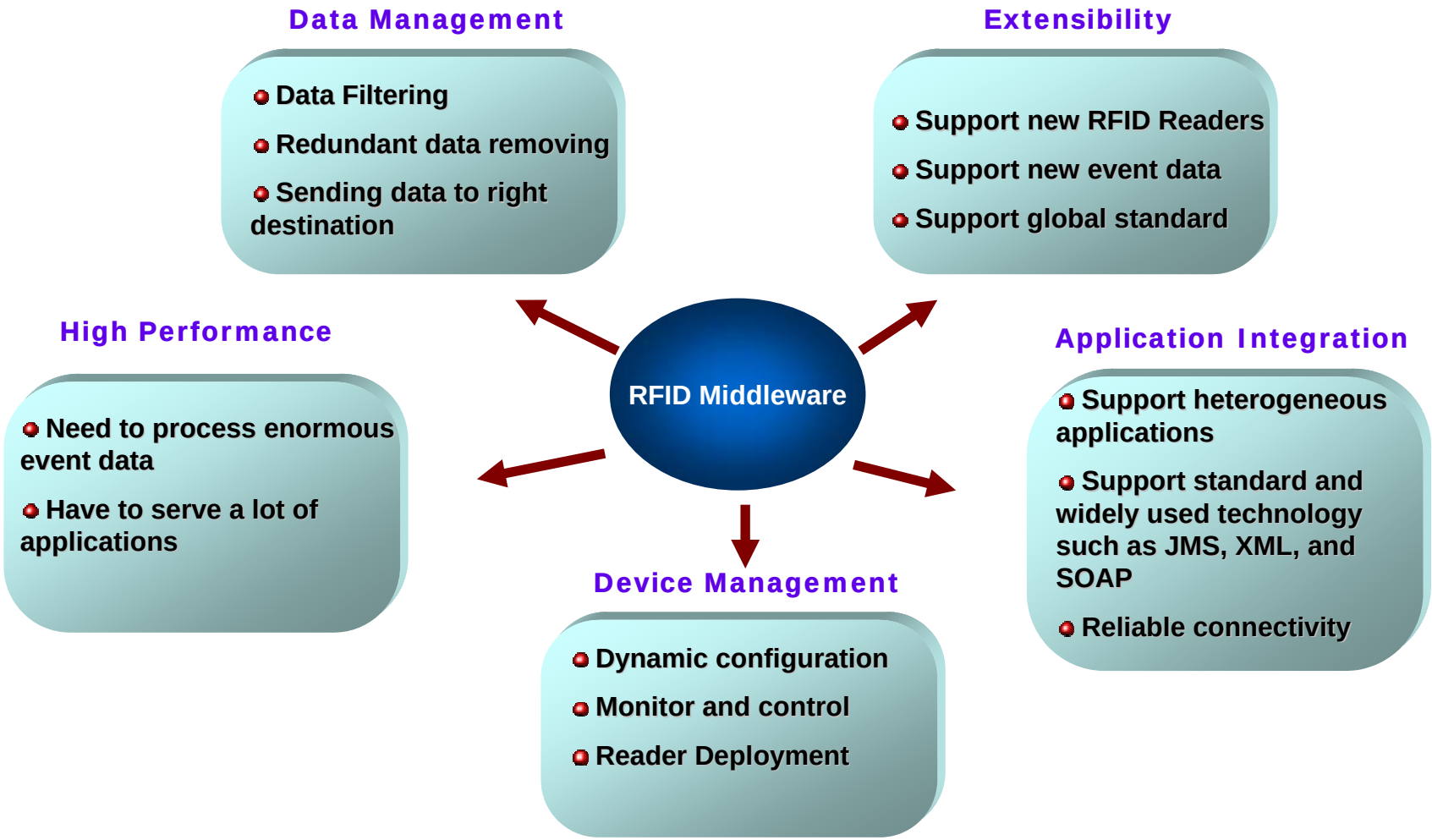


Introduction

- **RFID Middleware**
  - Process real time event data



Requirements of RFID Middleware

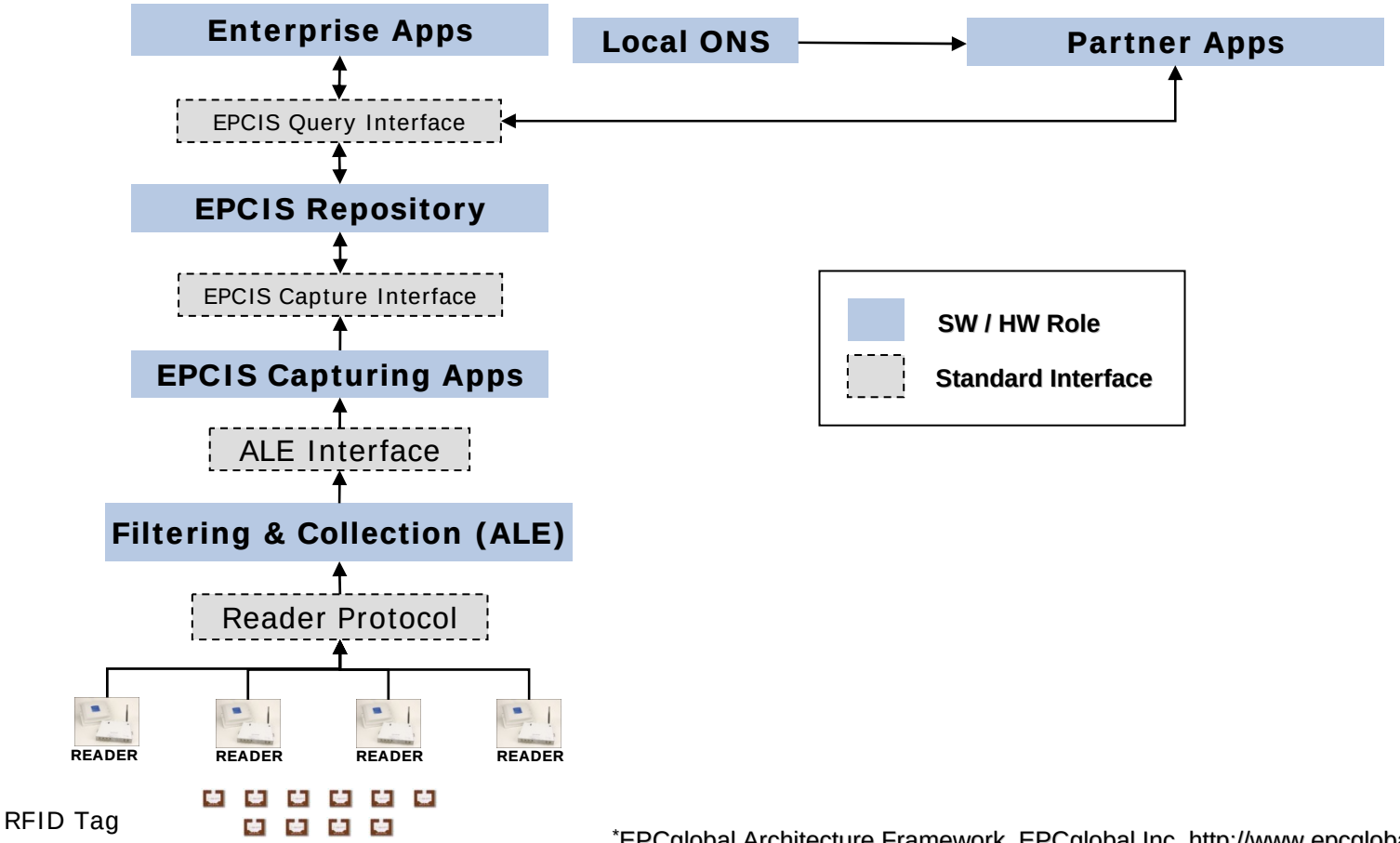


Standard for RFID Middleware

- ☑ **EPCglobal Standard**
  - Joint venture between EAN international and UCC Inc.
  - Developed by Auto-ID center
  - Objectives
    - Achieving worldwide adoption and standardization of EPC technology
  - Suggests
    - EPC network architecture for RFID technology
      - Standard for RFID Middleware
    - Roles and Interfaces
    - Standard Data format
  - Benefits
    - True visibility of items in supply chain
    - More accurate and immediate information

Standard for RFID Middleware

☑ Overview of EPC Network Architecture*



*EPCglobal Architecture Framework, EPCglobal Inc. http://www.epcglobalinc.org

Standard for RFID Middleware

- **Roles of Application Level Events (ALE)**
  - Receiving EPCs from one or more data sources
  - Accumulating, filtering and grouping EPCs
  - Reporting in various forms

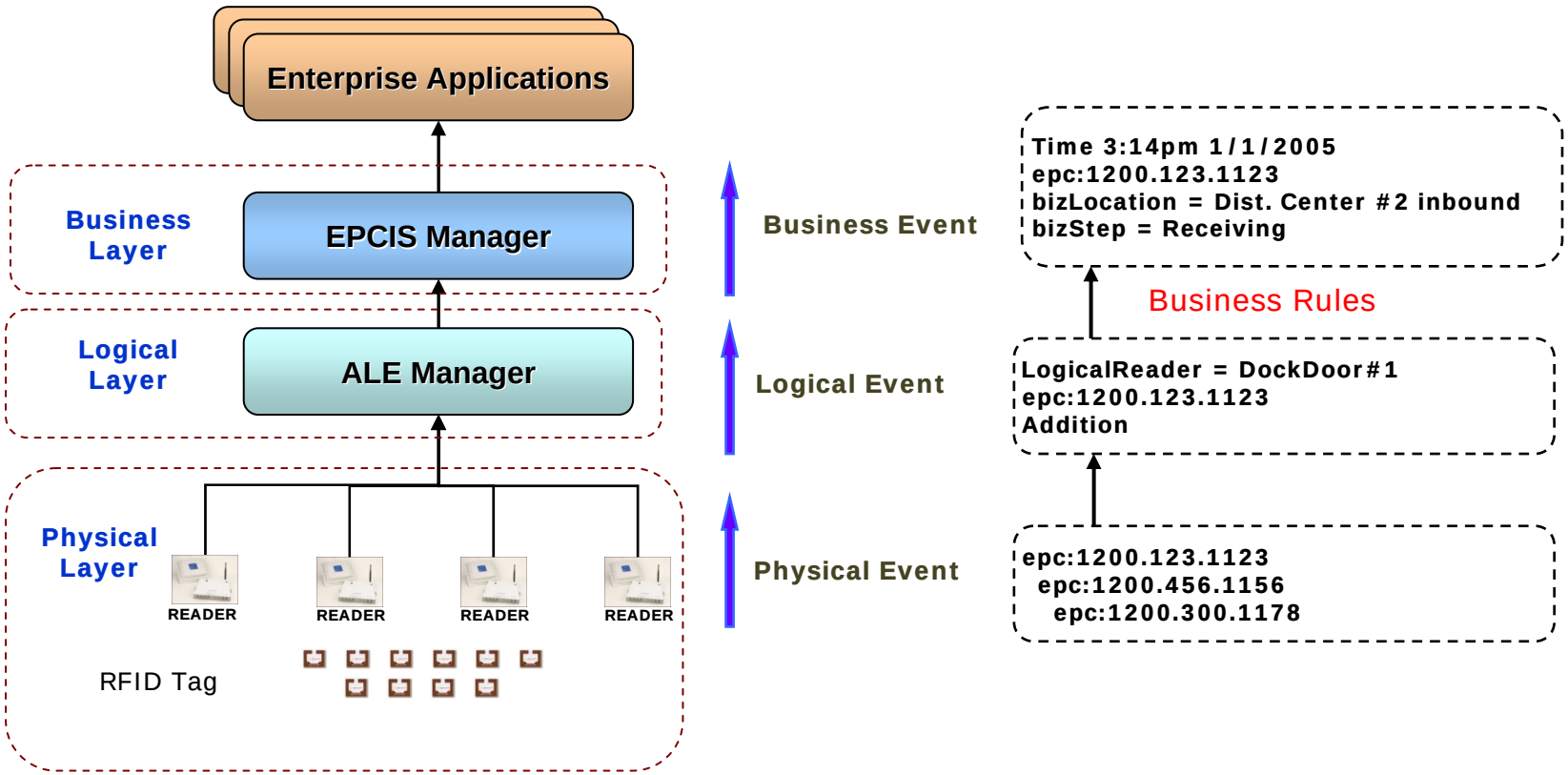
- **Roles of EPC Information Services (EPCIS)**
  - Store biz events
  - Respond queries from various clients
  - Provides a uniform programmatic interface to various clients

Overview of LIT Middleware

- **Objectives of LIT Middleware**
  - To provide highest level of extensibility
  - To provide fast & accurate data collection
  - To support heterogeneous applications
  - To support diverse logistics devices
- **Layered Architecture**
  - Physical Layer
    - RFID tag & Reader
  - Logical Layer
    - ALE Manager
      - Standard reader interface and reader integration module
      - Smoothing filter & Continuous Query Processor
  - Business Layer
    - EPCIS Manager
      - Data Source Access Component
      - Continuous query processor
      - Track & Trace query module

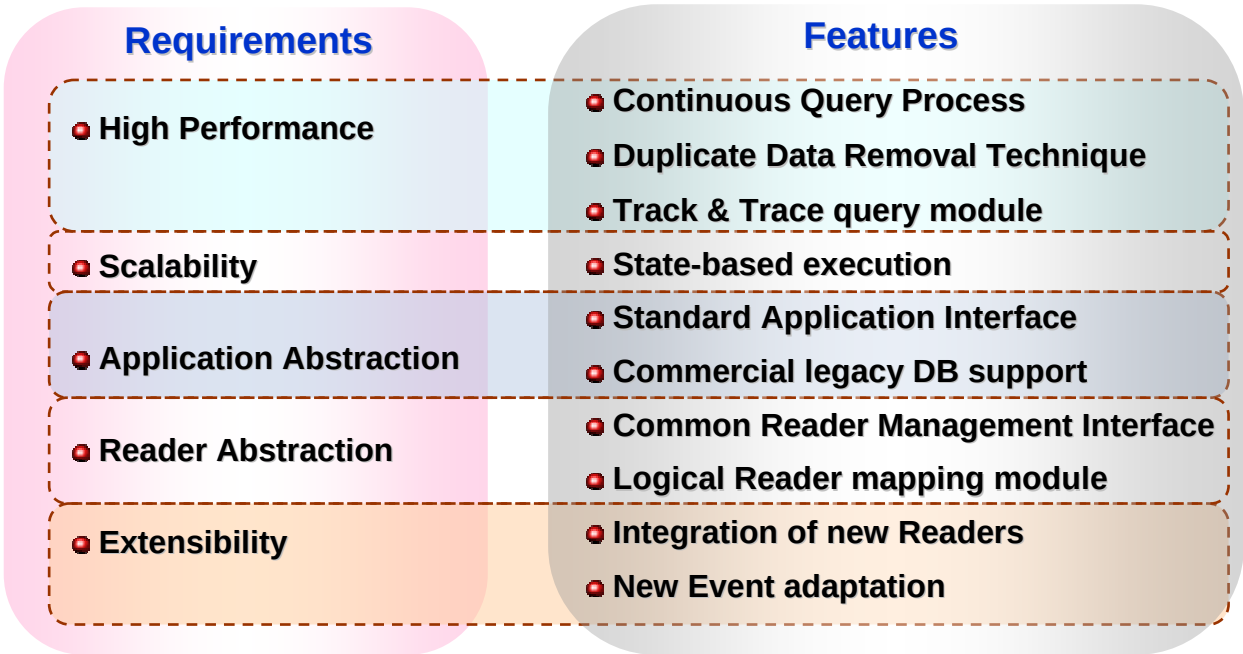
Overview of LIT Middleware

☑ Top level Architecture of LIT Middleware



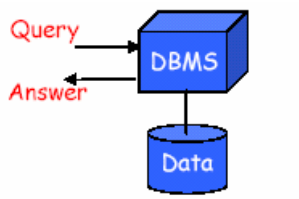
Design concepts of LIT Middleware

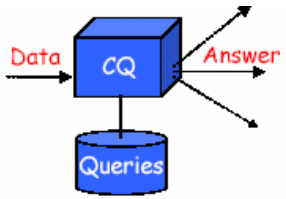
- **Features from Requirements**
  - Analyzed Requirements of Middleware
  - Analyzed EPCglobal Standard
  - Deduced Features of Middleware

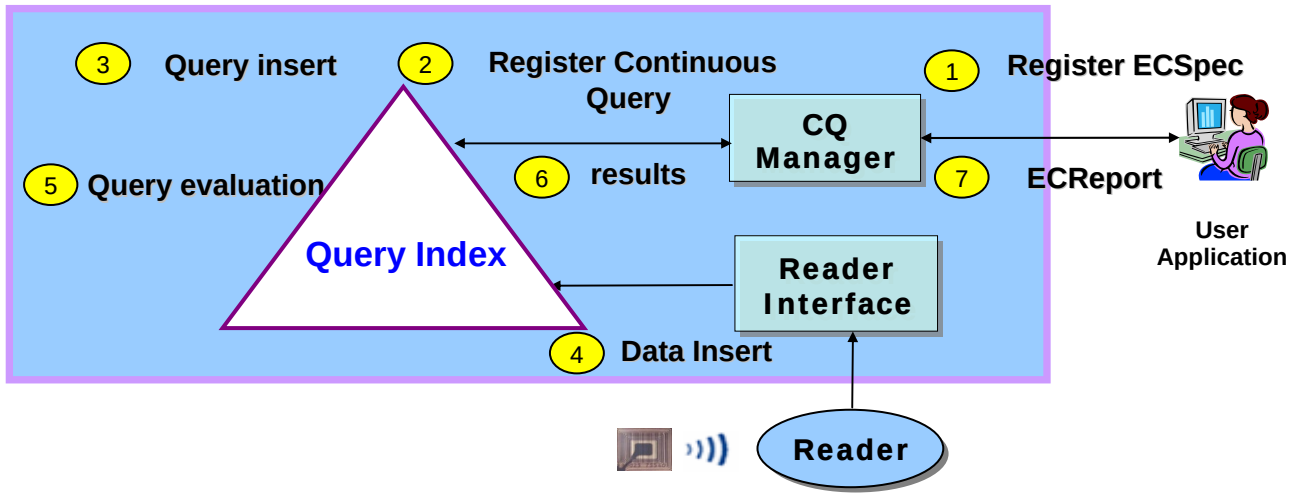


Features of ALE Manager

- **High performance**
  - Continuous Query Process
    - RFID data are gathered continuously by many readers
    - EPCglobal mentions standard specification (ECSpec) for filtering and collecting data during fixed time interval
    - Query Index – representing query using spatial data structure

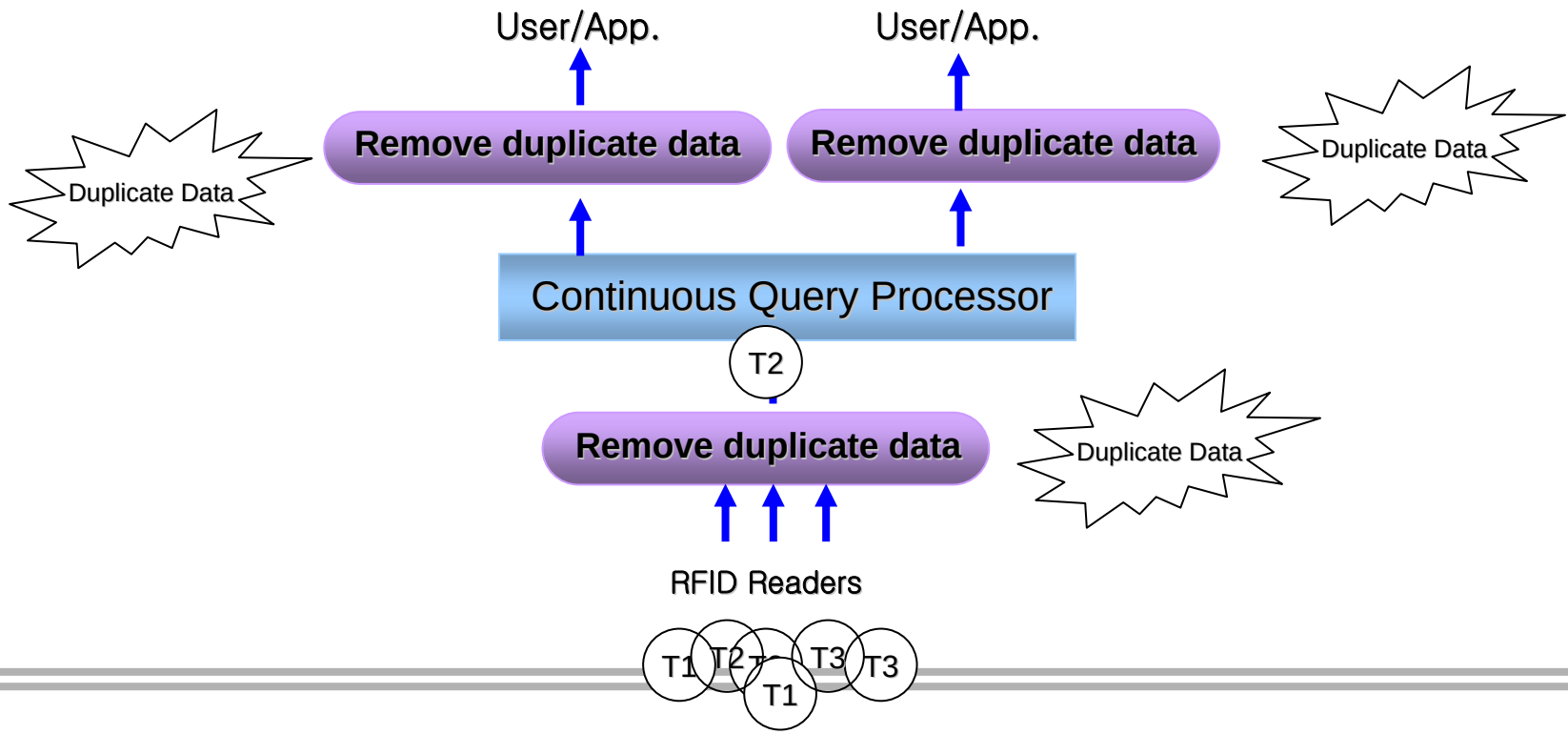






Features of ALE Manager

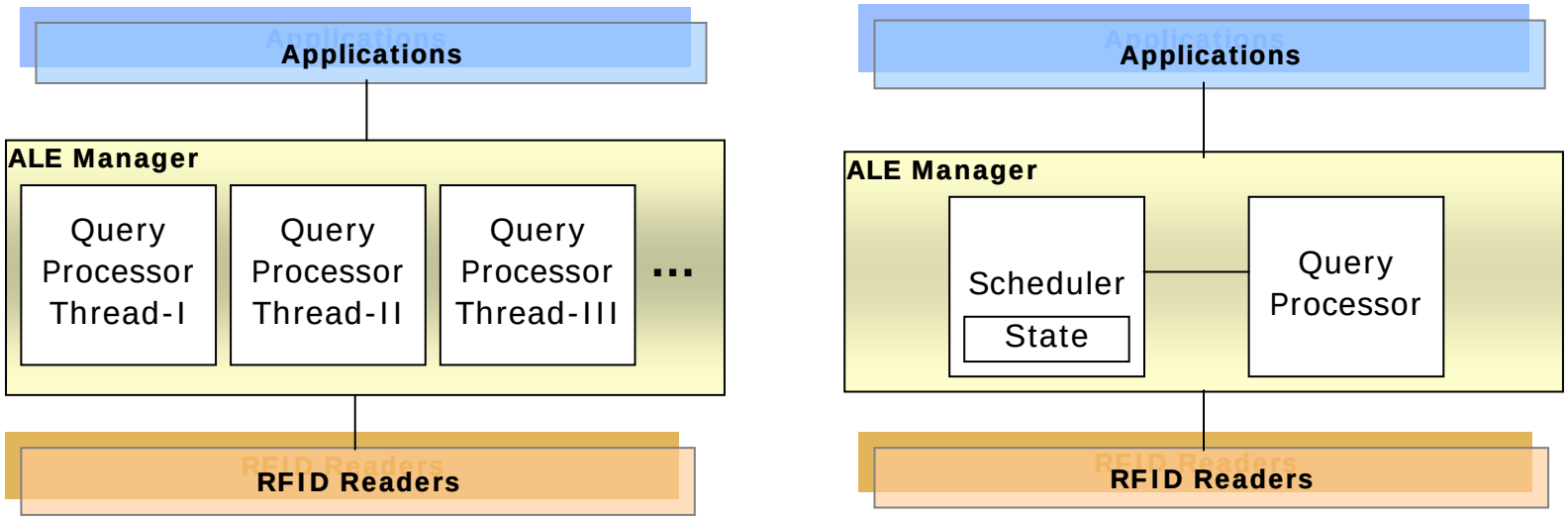
- ☑ **High performance**
  - ➢ Duplicate data removal technique
    - ● Time duration of each query called event cycle.
    - ● During event cycle reader may read same tag many times



Features of ALE Manager

☑ Scalability

- Thread-based execution VS. State-based execution*
- Thread-based execution model incurs significant overhead due to
  - Cache misses
  - Lock contention
  - Switching
- State-based execution model
  - Single scheduler
  - Avoids the limitation of thread-based model



*Operator Scheduling in a Data Stream Manager. *29th VLDB Conference, Berlin, Germany, pp 838-849*

Features of ALE Manager

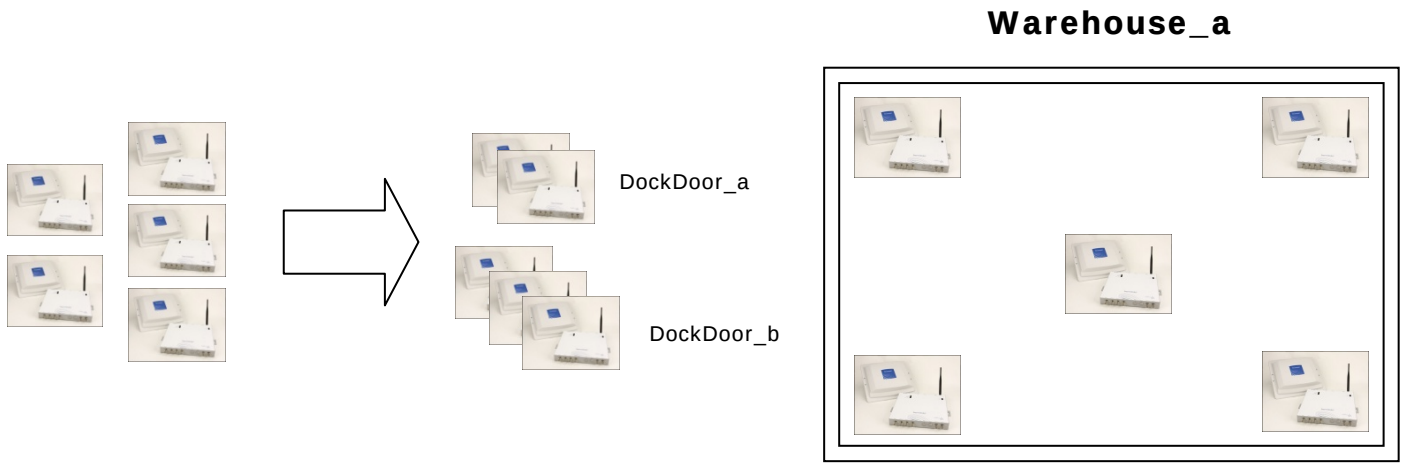
- ☑ **Application Abstraction**
  - ➢ Provides independence from different types of applications
  - ➢ EPCglobal proposed extensible application interface
    - ● API for Management
      - – API for Reader Management
      - – API for Middleware Management
    - ● API for Query
      - – Define, Undefine, Subscribe, Unsubscribe
      - – Immediate, Poll
  - ➢ Common Application Query & Response (by EPCglobal)
    - ● ECSpec, ECReports

Features of ALE Manager

☑ Reader Abstraction

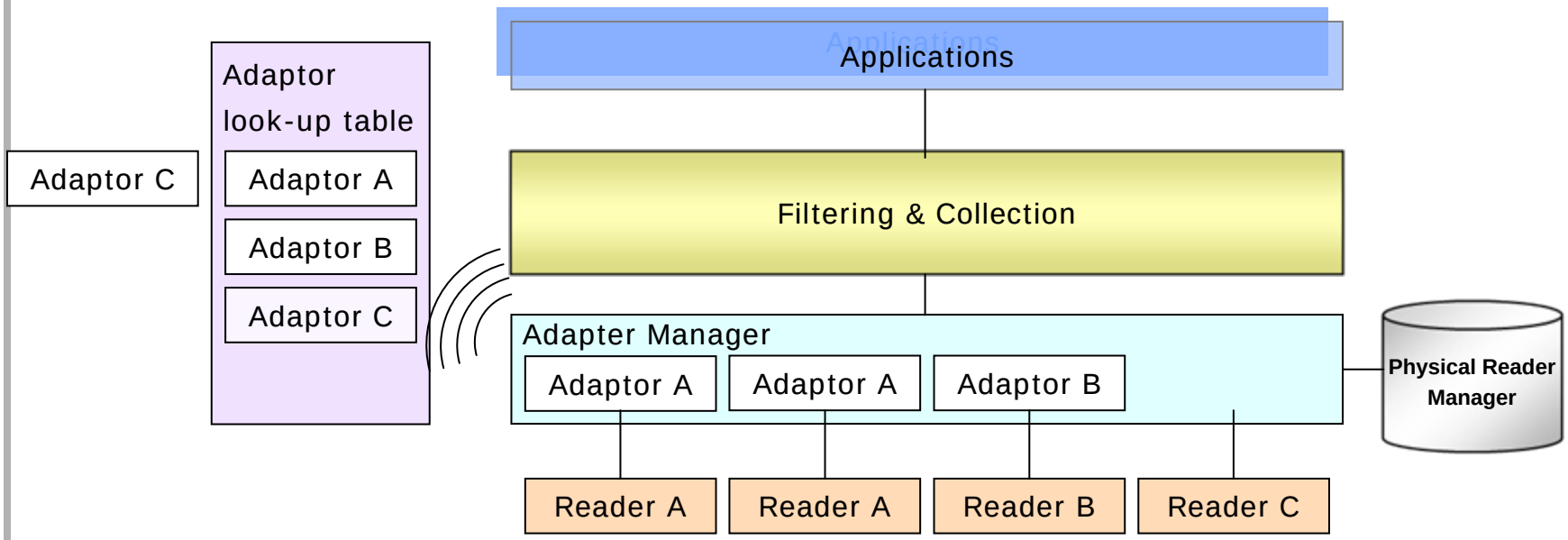
- Common Reader Management Interface
  - Manage heterogeneous RFID Reader
  - Using EPCglobal Reader Protocol standard

- Logical Reader Mapping
  - Independent from
    - Number of RFID Readers
    - Arrangement of RFID Readers



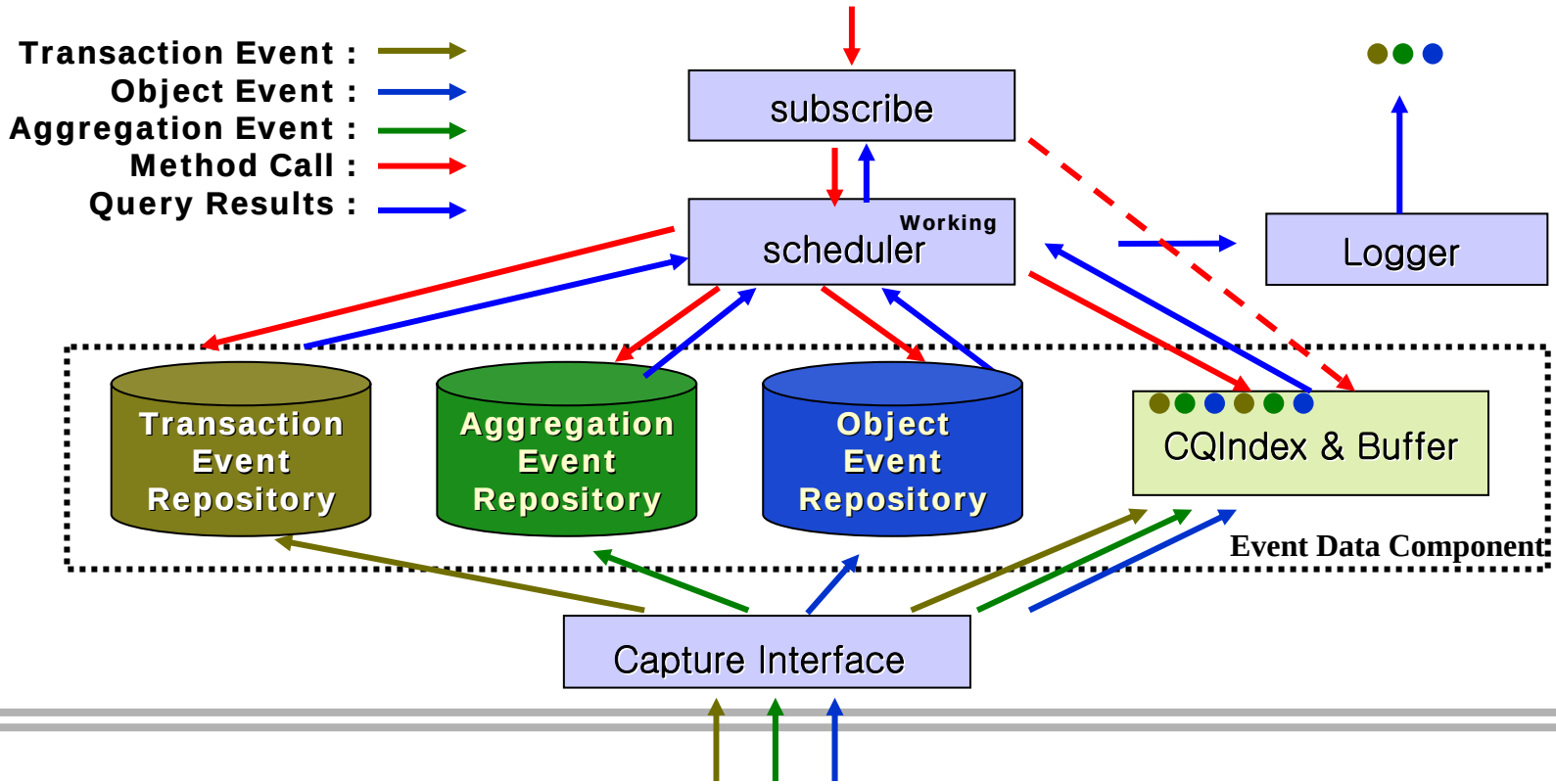
Features of ALE Manager

- **Reader Extensibility**
  - To support different types of RFID Readers from various vendors
  - UNIX Device Driver, Jini architecture
  - Adaptor look-up table



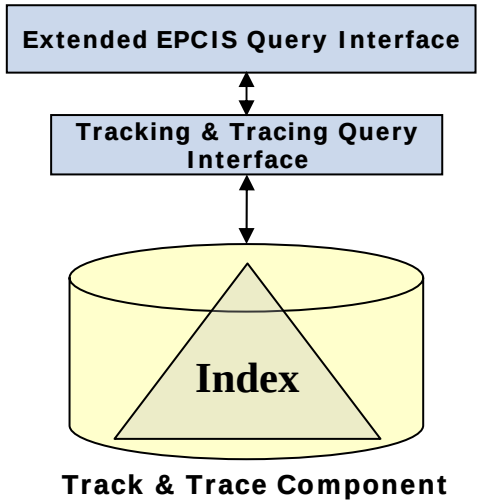
Features of EPCIS Manager

- ☑ **High performance and scalability**
  - ➢ Efficient Subscribe operation (using CQ Index technique)
    - Reduce processing time
    - Only first time access repository



Features of EPCIS Manager

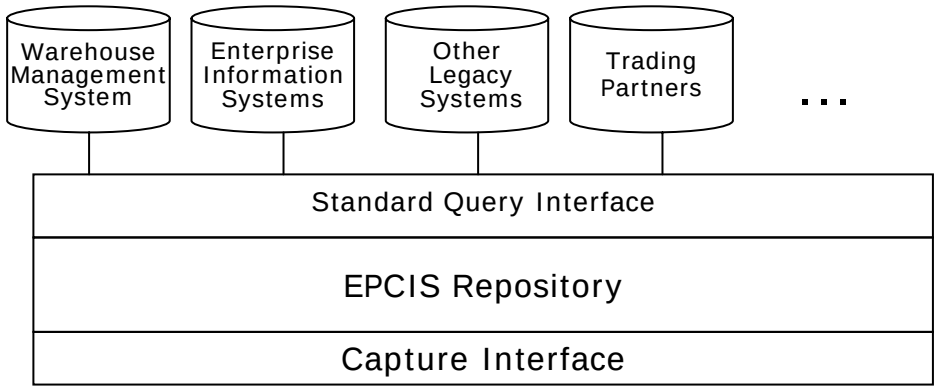
- ☑ **High performance**
  - ➢ Spatial index query for track & trace RFID tag object
  - ➢ Example of query
    - ● Where is the product #1? (present query)
    - ● Which products left warehouse B two hours ago? (past query)
  - ➢ Extended query interface
  - ➢ Fixed Interval R-tree*



*Design and Implementation of an Index Structure Using Fixed Intervals for Tracing of RFID Tags. *ICCSA, LNCS 3981, pp175-185*

Features of EPCIS Manager

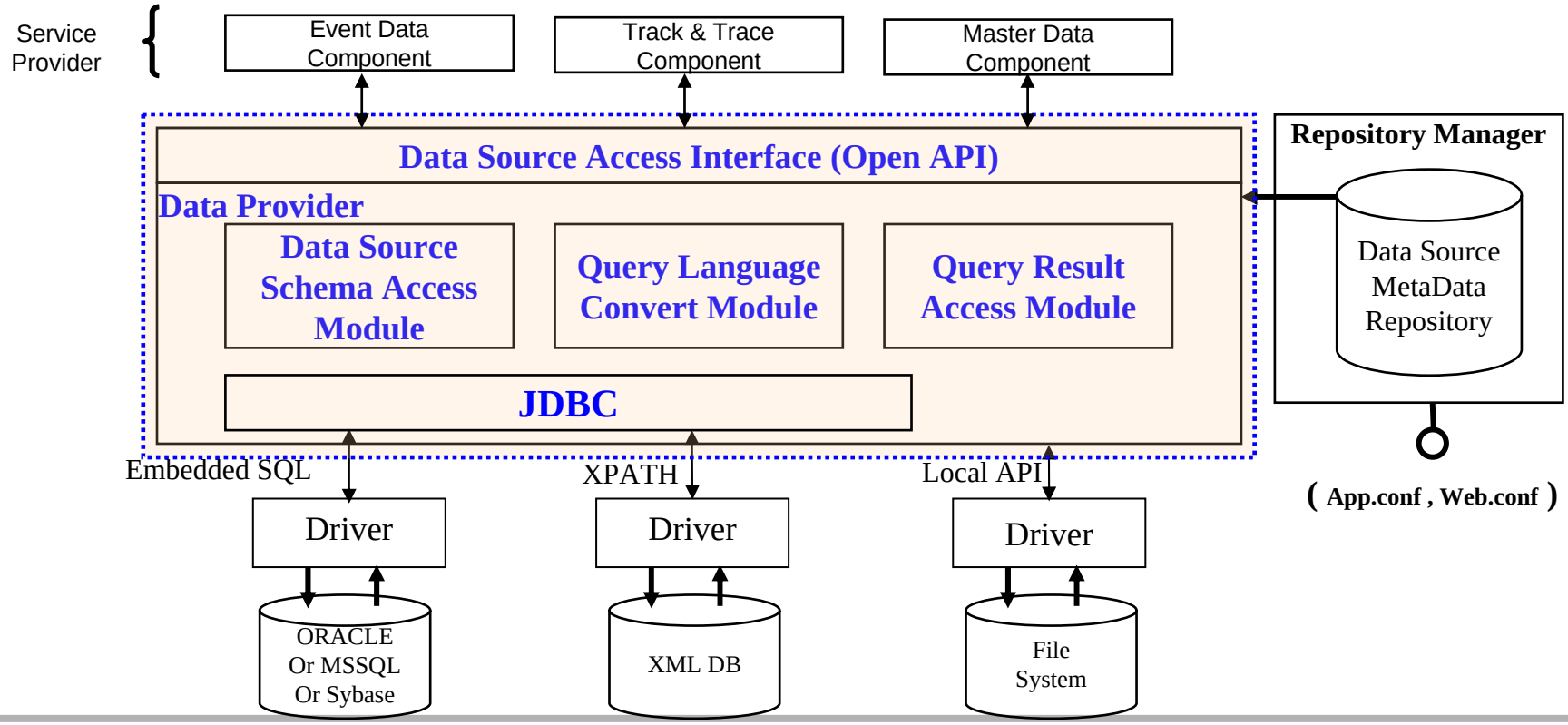
- ☑ **Application Abstraction**
  - ➢ Independent from types of applications
  - ➢ Persistent & compatible to store different types of EPCIS event.
  - ➢ Standard Query Interface (proposed by EPCglobal)
    - Standard data format
      - XML
    - Standard communication protocols : Query/Capture Service
      - HTTP
      - TCP
      - Web Service (wsdl)



Features of EPCIS Manager

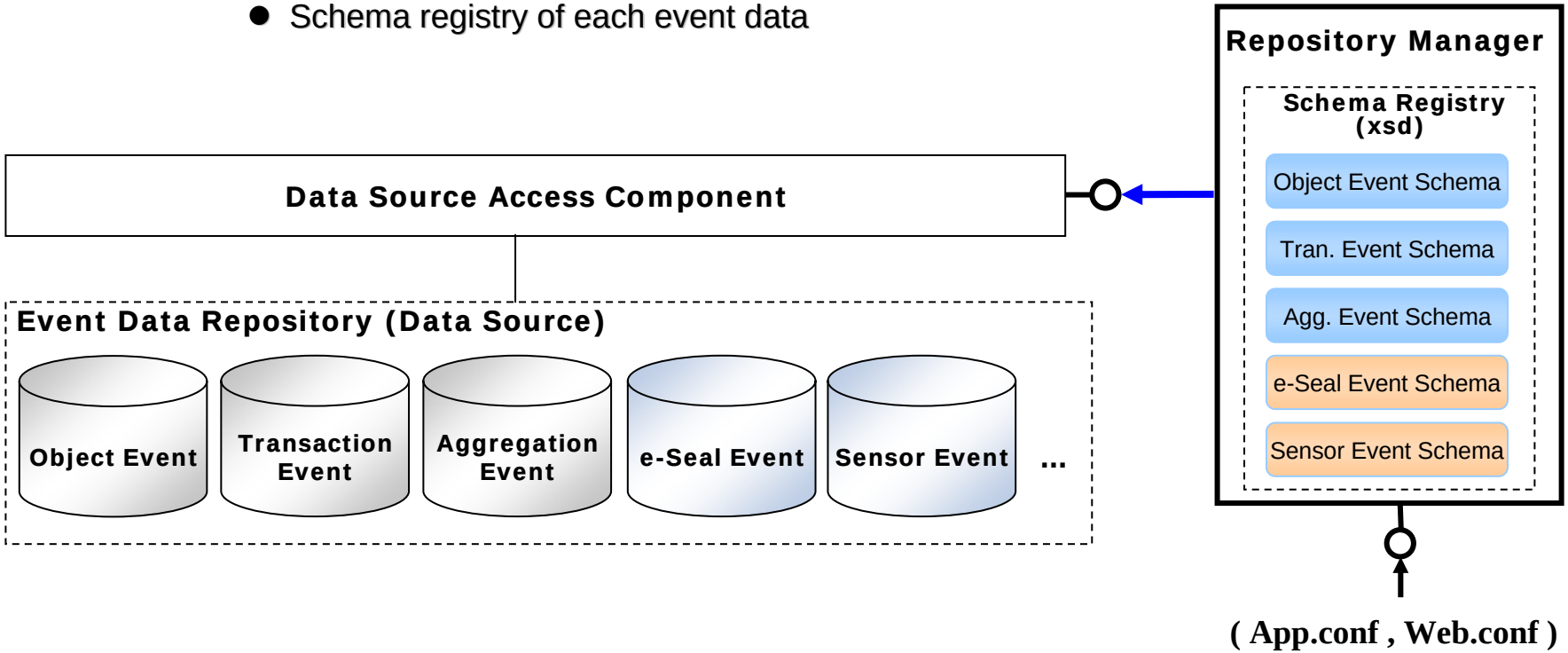
☑ Application Abstraction

- To support commercial legacy DB (Oracle, MSSQL…), XML DB and File system
  - We designed Data Source Access Component

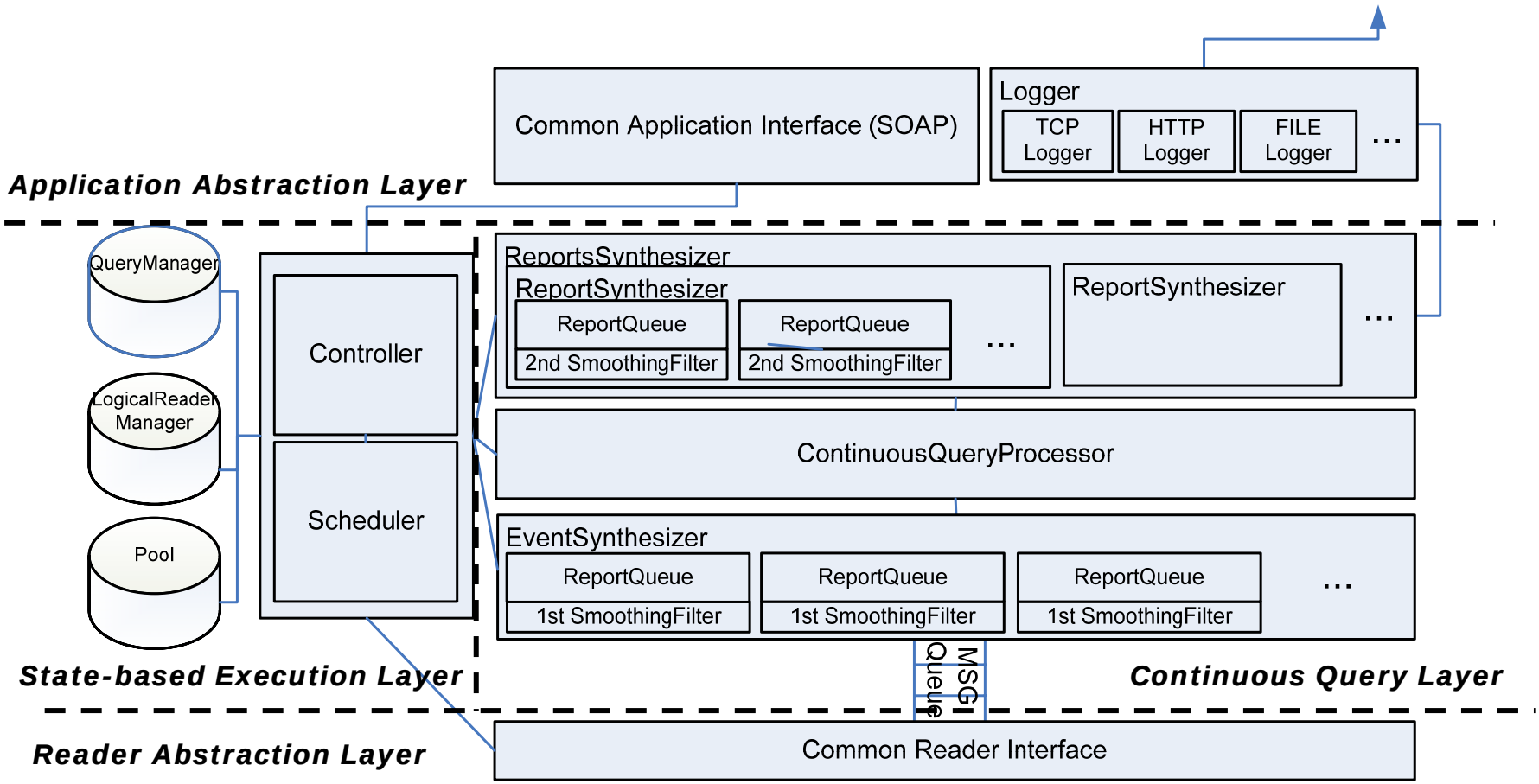


Features of EPCIS Manager

- ☑ **Extensibility**
  - To support new EPCIS Event
    - Ex: Sensor Event , e-Seal Event etc.
    - XSD, WSDL file extension.
  - Repository Management Module
    - Schema registry of each event data



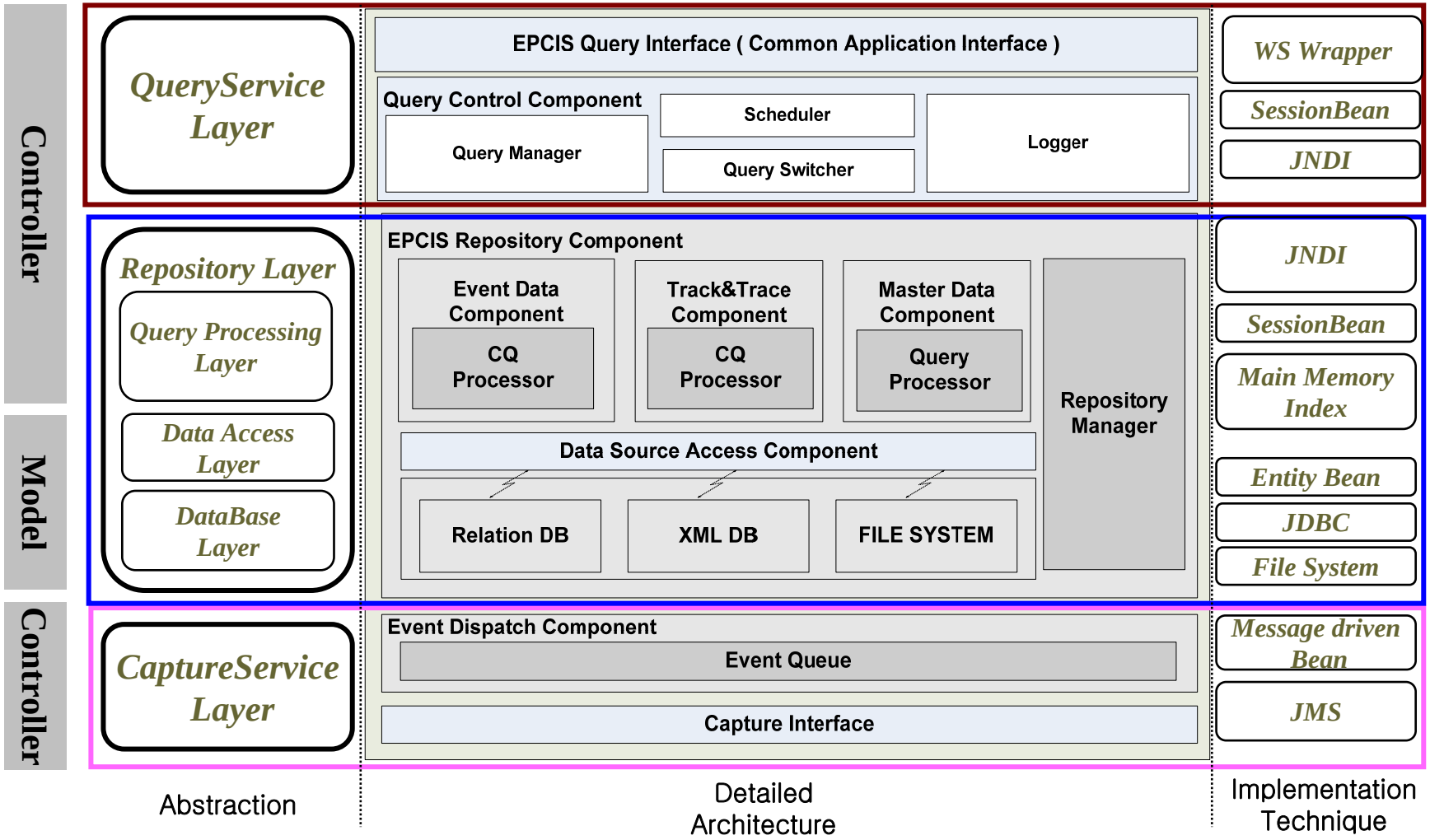
Software Architecture of ALE Manager



Software Architecture of ALE Manager

- **Reader Abstraction Layer**
  - Support different types of Readers from various vendors
  - Can integrate new Readers at runtime
- **State-based execution and Continuous Query Layer**
  - Control and manage ALE
  - CQ processor is implemented using query index
    - To support query defined by application
  - Query manager stores and manages query information
  - Reader manager keeps information of logical and physical reader mapping
  - Smoothing filter is used to remove duplicate data
- **Application Abstraction Layer**
  - Provides RFID data access by Standard Interface
  - Used standard protocol
    - TCP, HTTP, FTP, WSDL.
  - Support standard format (XML)

Software Architecture of EPCIS Manager



Software Architecture of EPCIS Manager

- **Capture Service Layer**
  - Provides standard interface
    - To capture events from EPCIS capturing application
  - Support standard format (XML)

- **Repository Layer**
  - Continuous Query processor for efficient subscribe operation
    - Implemented using query index
  - Implemented Fixed Interval R-tree for Track & Trace query
  - Implemented Data Source Access component
    - To support File system, Legacy DB and XML DB.

- **Query Service Layer**
  - Control and manage query processing
  - Provides Standard query interface
  - Used standard protocol
    - HTTP, WSDL

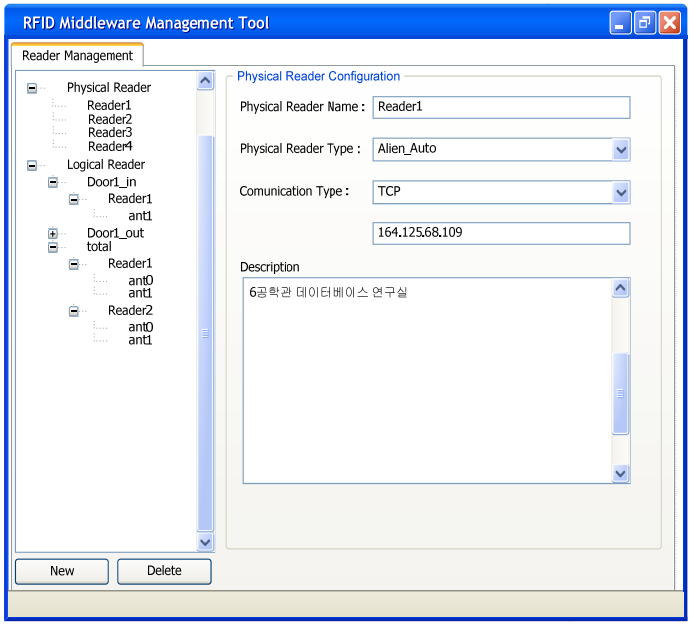
Implementation Environment

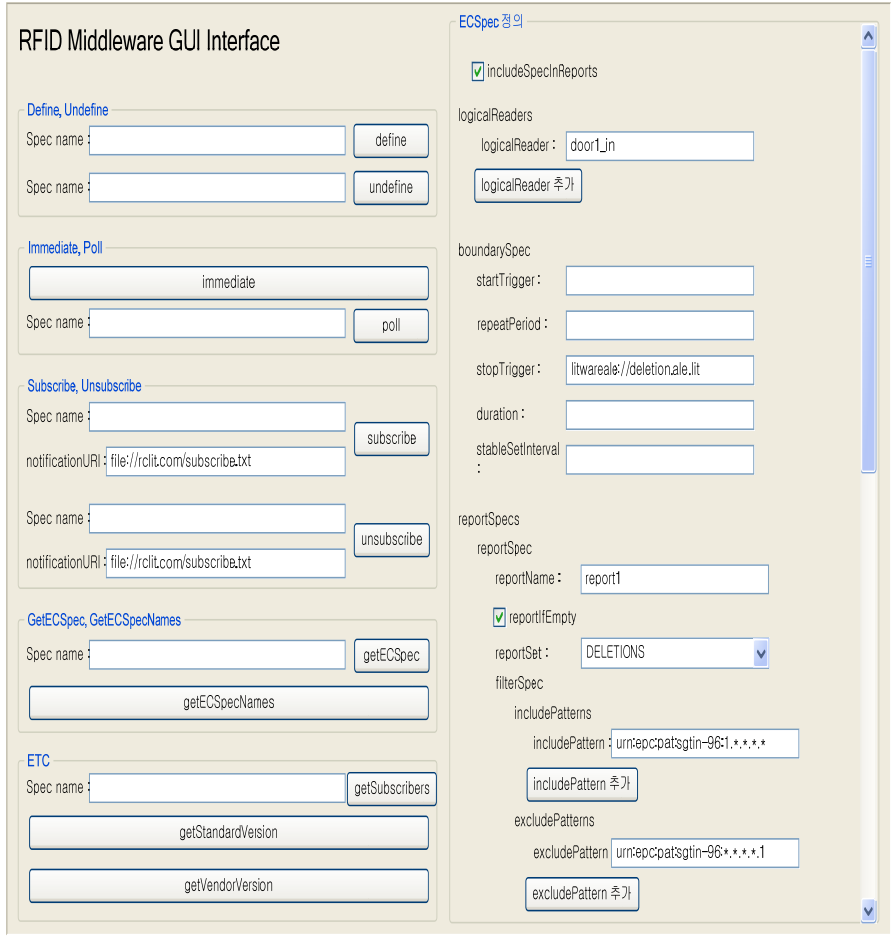
- **CPU:** Pentium IV 2.66 GHz.
- **RAM:** 1 GB.
- **OS:** Windows XP Pro. SP2
- **Language:** JAVA (JDK 5.0).
- **Web server:** SUN Application server.
- **RFID Reader, Antenna and Tag:**
  - Alien, Intermec, Kiscomm, KPC, LIT, Thingmagic.

Implementation

☑ Implemented software

- RFID middleware
- RFID Reader Management Tool
  - Manages physical RFID readers
  - Manages logical readers
- GUI for ALE Manager Client
  - Common Application Interface





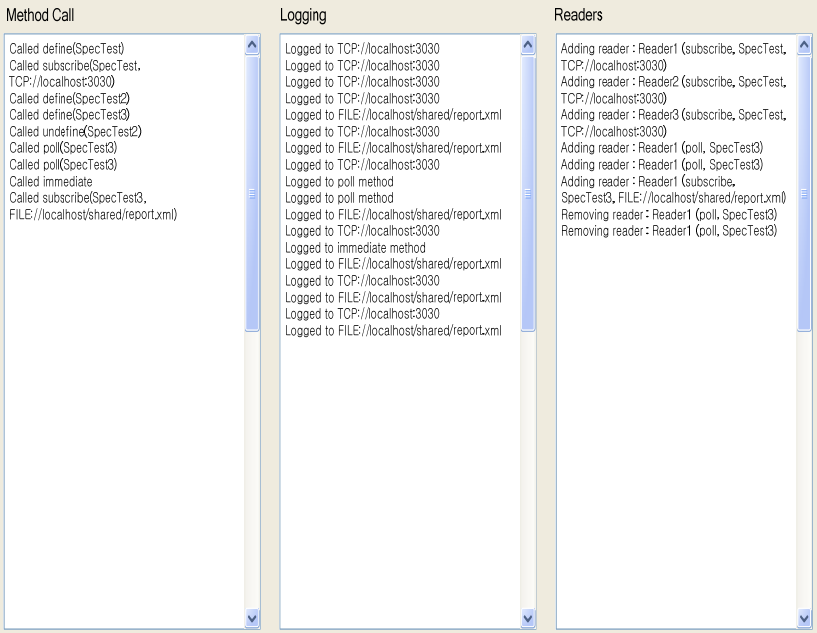
Implementation

☑ Implemented software

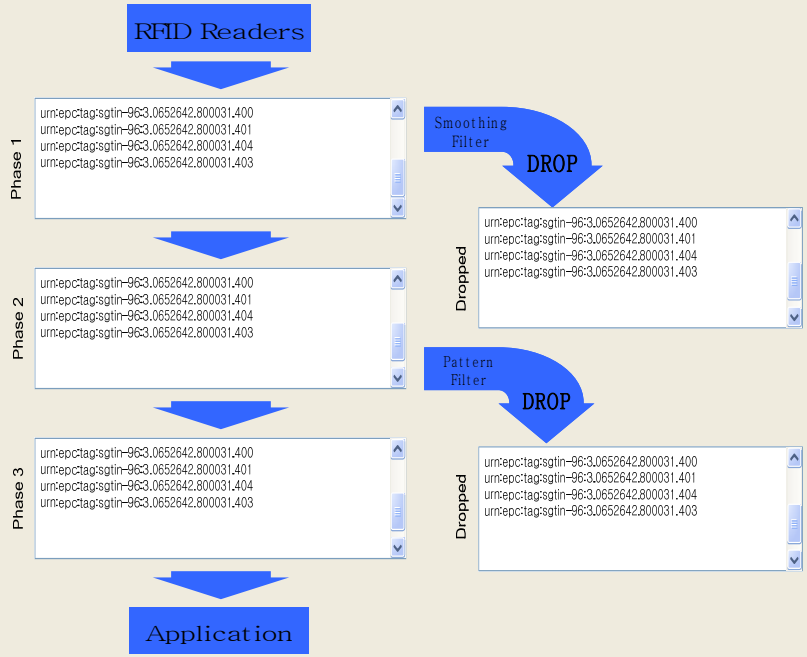
➢ RFID Middleware Monitor

- Monitor RFID tag data
- Monitor RFID middleware interface call

RFID Middleware Monitor - Interface



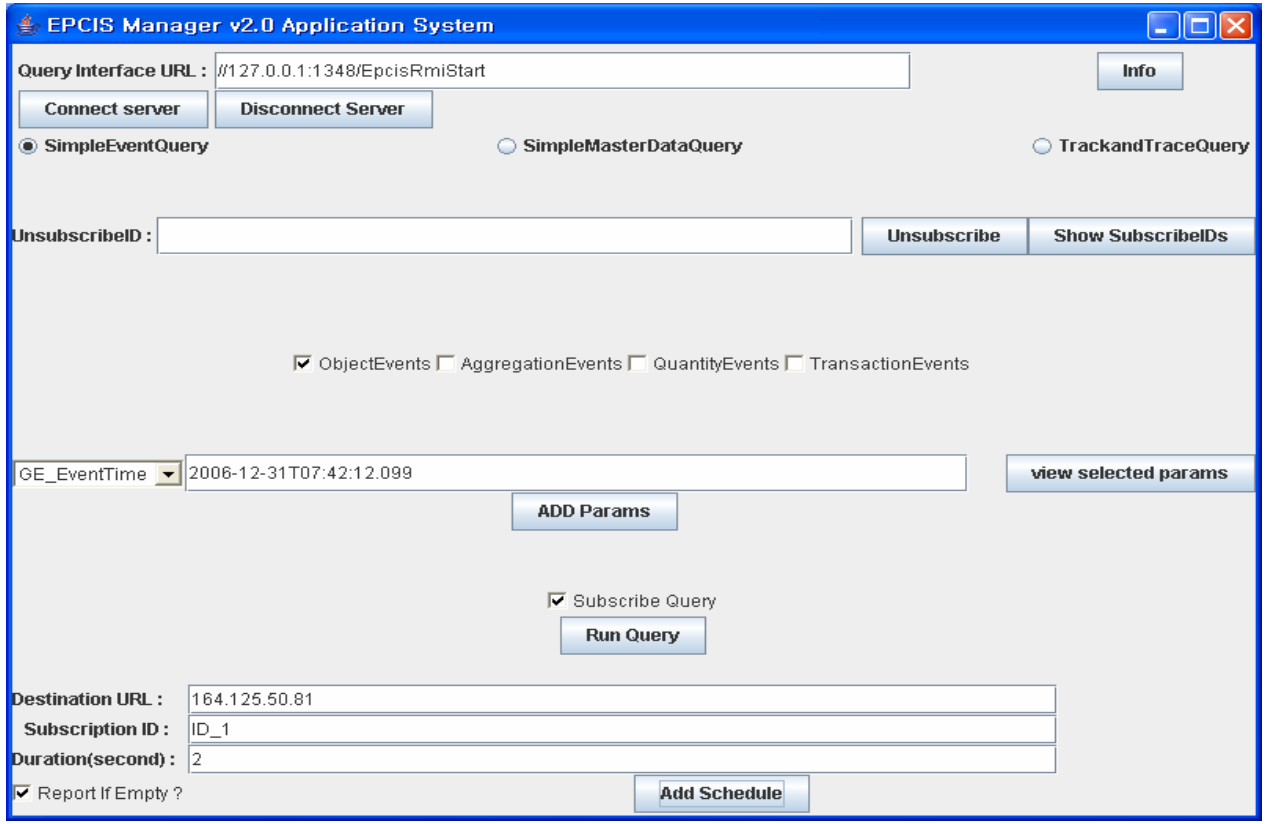
RFID Middleware Monitor - Tags



Implementation

☑ Implemented Software

- GUI for EPCIS Manager Client



Conclusion & Future work

☑ Conclusion

- Studied EPCglobal Network Architecture.
- Analyzed requirements of RFID Middleware
- Design Middleware based on EPCglobal standard
  - Layered Architecture
  - Provides important features such as High performance, Scalability, Abstraction and Extensibility
- Implement Middleware with features
  - User friendly GUI
  - Real time monitoring

☑ Future work

- Requirements analysis and implementation of Middleware to support
  - RFID, RTLS (Real Time Locating System) and Sensor data

Thanks for your attention

Questions?

Ashad Kabir
makcse@pusan.ac.kr