Design and Implementation of Pedigree System for RFID Application

Dept. of Logistics Information Technology
Pusan National University.
Mahbubur Rahman

Aug 28, 2009
Contents

- Introduction
- Related Work
  - Pedigree
  - EPCIS
- Pedigree Scenario with EPCIS Event
- Pedigree System Design Issue
- Pedigree System
  - Design
  - Implementation
- Conclusion & Further Research
Introduction

◆ A Genuine Product Business

→ Consumers want to know whether the expensive good is genuine or not
  ▪ Distribution of Genuine Product From Manufacturer to Consumer
  ▪ Using the Non-counterfeit Equipment
    ▪ Hologram
    ▪ non-counterfeit Label
    ▪ non-counterfeit Cap

Is this good a Genuine Product?

<Non-counterfeit Equipments>

[Hologram] [non-counterfeit Label] [non-counterfeit Cap]
Introduction

◆ **Distrust** about the Genuine Product
  ➔ Sales of Counterfeit Product (ex. Alcohol, Drug, Honey...)
  ➔ What is worse, **Counterfeit of Non-counterfeit Equipment**
Related Work - Pedigree

What is Pedigree?

- A **certified record** that contains information about **each distribution**

Goal

- To help ensure **only authentic products are distributed**
- A step in securing the supply chain

It contains information of

- Product
- Transaction
- Distributor
- Recipient
- Signatures

Related Work - Pedigree

◆ Requirements of Pedigree

 ➤ Tracking Products
  ▪ RFID tags are integrated into product
  ▪ communicate unique data about the item as they move through the supply chain

 ➤ Recording and Managing Data
  ▪ Each distributor maintains complete, accurate and secure records of the pedigrees
  ▪ Distributors must manage and report data

 ➤ Communicating the Data with Partners
  ▪ Partners need a secure environment
  ▪ share information about pedigrees

SupplyScape, Drug Pedigree Requirements, 2007
Security Assessment of EPCglobal Architecture Framework: Auto-ID Labs White Paper WP-SWNET-017
Related Work - Pedigree

◆ Pedigree Format

→ Pedigree
  ▪ A wrapper element
  ▪ The pre-existing pedigree for an item is wrapped inside of a pedigree layer element
  ▪ ("shippedPedigree" or "receivedPedigree") + “Signature”

→ initialPedigree
  ▪ The innermost component of pedigree
  ▪ Initiated by manufacturer
  ▪ Consist of productInfo, itemInfo

→ shippedPedigree
  ▪ Represent a shipping stage
  ▪ Add in every shipping stage
  ▪ Consist of transactionInfo, senderInfo, recipientInfo, signatureInfo

→ receivedPedigree
  ▪ Represent a receiving stage
  ▪ Add in every receiving stage
  ▪ Consist of receivingInfo, signatureInfo
Related Work - EPCIS

◆ EPCglobal Network Architecture
  ➔ RFID Reader
    ▪ Observations of RFID Tag
  ➔ ALE
    ▪ Filtering & Collection
  ➔ EPCIS Capturing Application
    ▪ Make business event
  ➔ EPCIS(EPC Information Services)
    ▪ Store & Search EPCIS-level events
  ➔ ONS
    ▪ Search manufacture EPCIS service address
  ➔ EPCDS
    ▪ Search list of EPCIS service address

Related Work - EPCIS

- EPCIS Business Event Creating Process

**EPCIS (EPC Information Service)**
- RFID Event DB
- Sensor Event DB
- RTLS Event DB
- Pedigree DB

**Capturing App.**

**ALE (Collection & Filtering)**
- RFID M/W
- Sensor M/W
- RTLS M/W

**Logical Event**

**Physical Event**
- Tag ID, Reader ID, Sensing info.,
- Active Tag
- Passive Tag
- Sensor Tag
- Tag ID, Tag Location,

**Business Event**
Related Work - EPCIS

◆ EPCIS Event Data
  ➔ EPCIS Event
    ▪ a common base type for all EPCIS events
  ➔ Object Event
    ▪ represent observations of EPCs
  ➔ Aggregation Event
    ▪ represent for aggregations
  ➔ Quantity Event
    ▪ represent for quantity
  ➔ Transaction Event
    ▪ represent for business transactions
    ▪ **Proper to represent of Pedigree**
      ▪ Attach to extension point

Reference: EPCglobal, "EPCIS Standard v.1.0.1", September, 2007
**Related Work - Pedigree**

- **Simplified pedigree process**

![Diagram showing the simplified pedigree process involving different entities such as Manufacturer, Wholesaler, Retailer, and Consumer. Each entity exchanges Product Info, Sender A Signature, and Recipient B Signature, maintaining a pedigree record.]
Pedigree System Design Issue

- EPCIS User want to extend EPCIS for Using Pedigree
  ➔ Need to Extend EPCIS for Pedigree Easily

- Customer want to know whether the product is genuine or not
  ➔ Need to provide the certain information to customers

How to Extend EPCIS for Pedigree Easily?
Pedigree System Design Issue

- **Easy to Extend EPCIS for Pedigree** by Using **Pedigree Library**
  - Capturing App: Create Business Event
    - Create Transaction Event
    - Create Pedigree
  - **Pedigree Library**
    - Provide **Pedigree Processing API**
      - Create Pedigree
        - Receive Partner’s Pedigree
        - Add Current Pedigree
        - Add Signature
  - **Advantage**
    - Independent on Application
    - Easy Pedigree Extension
Pedigree System Design Issue

◆ Provide **Pedigree Library** for Service Type
  ➤ Pedigree Service: Pedigree Creating Service
    ▪ Input: Information (XML), Certificate
    ▪ Output: Pedigree
  ➤ Advantage
    ▪ No need to implement for Pedigree Creating in Every Capturing App.
Pedigree System Design Issue

- Verification of the Genuine Transaction
  The Consumer Kill the Product Serial Number

<table>
<thead>
<tr>
<th>Product Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProductName: ABCAlcohol</td>
</tr>
<tr>
<td>ProductionDate: 09.01.02</td>
</tr>
<tr>
<td>ProductCode: 0614141.000000001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transaction History</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ABCManufacturer, KyungKi SiheungSi 343-2, Kim Ji Won Shipping, 2009.01.03</td>
</tr>
<tr>
<td>2. K-Wholesaler, Busan Kimhae 22-1, Park Mi Kyung Receiving, 2009.01.04</td>
</tr>
<tr>
<td>3. K-Wholesaler, Busan Kimhae 22-1, Kim Je won Shipping, 2009.01.05</td>
</tr>
<tr>
<td>4. L-Retailer, Busan KeumjungGu 443-1, Lee Su Han Receiving, 2009.01.06</td>
</tr>
</tbody>
</table>

If genuine
- This item serial number is available.
- Will you buy it?
  - Yes
  - No

If forgery
- This item serial number is not available.
- It is sold at '2009-01-08'.
- This product may be forgery!
Pedigree Scenario with EPCIS Event

**Scenario**

Wholesaler order 2 cases (10 Bottles in case) from Manufacturer
Pedigree Scenario with EPCIS Event

◆ Pre-Step: Register EPCIS Continuous Query at trading partners
  ➡ To enable Automatic Data Exchange of Shipping & Receiving Event
  ➡ Person in charge: Manufacturer, Wholesaler
  ➡ Using System
    ▪ Each Access App. & Each EPCIS
Pedigree Scenario with EPCIS Event

 Î Step 1: **Items Commissioning**

 ➤ Attach tags to items & Create Object Event for each item
 ➤ Create Initial Pedigree
 ➤ Person in charge: Manufacturer
 ➤ Using System
   - Manufacturer Capturing App. & EPCIS

---

**EPCIS Object Event**

- **EventTime:** 2008-06-25T00:01:00Z
- **EPCList**
  - EPC: urn:epc:id:sgtin:0614141.107340.1
  - EPC: urn:epc:id:sgtin:0614141.107340.10
  - ReadPoint: RP-1
  - Action: ADD
  - BizStep: commissioning

---

**Initial Pedigree**

- **serialNumber:** ip-1
  - **productInfo:** ABC Alcohol
  - **itemInfo**
    - expirationDate: 2008-06-30 18:00
    - quantity: 10
    - itemSerialNumber
    - EPC: urn:epc:id:sgtin:0614141.107340.1[1-10]
Pedigree Scenario with EPCIS Event

◆ **Step 2: Cases Commissioning**
  - Attach tags to cases & Create Object Event for each case
  - Person in charge: Manufacturer
  - Using System
    - Manufacturer Capturing App. & EPCIS

Case Commissioning

**EPCIS Object Event**

- **EventTime**: 2008-06-25T00:05:00Z
- **EPCList**
  - **EPC**: urn:epc:id:sscc:0614141.000000001
  - **ReadPoint**: RP-1
  - **Action**: ADD
  - **BizStep**: commissioning

Case EPC
Pedigree Scenario with EPCIS Event

◆ Step 3: **Pack Items to Case**
  - Create Aggregation Event
  - Person in charge: Manufacturer
  - Using System
    - Manufacturer Capturing App. & EPCIS

---

**EPCIS Aggregation Event**

- EventTime: `2008-06-25T00:10:00Z`
- ParentID: `urn:epc:id:sscc:0614141.000000001`
- ChildEPCs
  - EPC: `urn:epc:id:sgtin:0614141.107340.1`
  - EPC: `urn:epc:id:sgtin:0614141.107340.10`
- ReadPoint: `RP-1`
- Action: `ADD`
- BizStep: `item to case aggregation`
Pedigree Scenario with EPCIS Event

小康社会

Step 4: Pallet Commissioning

- Attach tags to pallets & Create Object Event for each pallet
- Person in charge: Manufacturer
- Using System
  - Manufacturer Capturing App. & EPCIS

---

EPCIS Object Event

- EventTime: 2008-06-25T00:15:00Z
- EPCList
- EPC: urn:epc:id:sscc:0614141.100000000
- ReadPoint: RP-2
- Action: ADD
- BizStep: commissioning

Pallet EPC

Pallet Commissioning
Step 5: Pack Cases to Pallet
- Create Aggregation Event
- Person in charge: Manufacturer
- Using System
  - Manufacturer Capturing App. & EPCIS

EPCIS Aggregation Event
- EventTime: 2008-06-25T00:20:00Z
- ParentID: urn:epc:id:sscc:0614141.100000000
- ChildEPCs
  - EPC: urn:epc:id:sscc:0614141.000000001
  - EPC: urn:epc:id:sscc:0614141.000000002
- ReadPoint: RP-2
- Action: ADD
- BizStep: case to pallet aggregation
Pedigree Scenario with EPCIS Event

◆ Step 6: **Receive Pallet at DC**
  - Create Object Event
  - Person in charge: Manufacturer
  - Using System
    - Manufacturer Capturing App. & EPCIS

**EPCIS Object Event**
- EventTime: **2008-06-25T00:25:00Z**
- EPCList
- EPC: `urn:epc:id:sscc:0614141.100000000`
- ReadPoint: RP-3
- Action: **OBSERVE**
- BizStep: **receiving**

![Receive Pallet at DC Diagram]
Pedigree Scenario with EPCIS Event

◆ Step 7: **Pick Cases (unpack pallet)**
  - Create Object Event
  - Person in charge: Manufacturer
  - Using System
    - Manufacturer Capturing App. & EPCIS

![Diagram of a person picking cases](image)

**EPCIS Object Event**
- **EventTime:** 2008-06-25T00:30:00Z
- **EPCList**
  - **EPC:** urn:epc:id:sscc:0614141.000000001
  - **EPC:** urn:epc:id:sscc:0614141.000000002
- **ReadPoint:** RP-3
- **Action:** OBSERVE
- **BizStep:** picking

**Case EPCs**
Pedigree Scenario with EPCIS Event

◆ Step 8: Manufacturer Ship Purchase Order to Wholesaler

 ➔ Create **Transaction Event**
   ▪ Include Pedigree Envelope

 ➔ Create **Pedigree Envelope**
   ▪ Create Container
   ▪ Create Pedigree
     ▪ Initial Pedigree (made in step 1)
     ▪ Create **Shipped Pedigree + Signature**

 ➔ Person in charge: Manufacturer

 ➔ Using System
   ▪ Manufacturer Capturing App. & EPCIS

**EPCIS Transaction Event**
- EventTime: **2006-06-25T07:16:00Z**
- EPCList
  - EPC: urn:epc:id:sscc:0614141.000000001
  - EPC: urn:epc:id:sscc:0614141.000000002
- ReadPoint: RP-4
- Action: ADD
- BizStep: shipping
- extention: PedigreeEnvelope

**PedigreeEnvelope**
- serialNumber: PE-1

**container**
- containerCode: urn:epc:id:sscc:0614141.000000001
- pedigreeHandle
  - pedigreeSerialnumber: P-1

**container**
- containerCode: urn:epc:id:sscc:0614141.000000002
- pedigreeHandle
  - pedigreeSerialnumber: P-2

**Pedigree**
- serialnumber: P-1
  - initialPedigree
  - shippedPedigree
    - serialNumber: sp-2
- senderInfo: www.Harim.com/EPCIS
- recipientInfo: www.Kimhae.com/EPCIS
- signatureInfo: Julia Roberts

**Pedigree**
- serialnumber: P-2
  - initialPedigree
  - shippedPedigree
Pedigree Scenario with EPCIS Event

**Step 9 : Manufacturer Publish Transaction Event (shipping)**

- EPCIS Continuous query
  - Event Based Query
  - Used to Obtain Transaction Event (shipping) is made in Step 8
  - Recipient Matching
- Person in charge : Manufacturer
- Using System
  - Manufacturer EPCIS

![Diagram showing the scenario]

**[Continuous Query]**
- Event Type: Transaction Event
- BizStep: shipping
- Recipient: Wholesaler

Send Transaction Event to Wholesaler (made in Step 8)
Pedigree Scenario with EPCIS Event

◆ **Step 10**: Wholesaler Receive Cases from Manufacturer

> Create **Transaction Event**
  - Include Update Pedigree Envelope

> Update **Pedigree Envelope**
  - Update **Pedigree**
    - Initial Pedigree (made in step 1)
    - Shipped Pedigree (made in step 8)
  - Create **Received Pedigree + Signature**

> Person in charge: Wholesaler

> Using System
  - Wholesaler Capturing App. & EPCIS

---

**EPCIS Transaction Event**

- **EventTime**: 2006-06-25T08:21:00Z
- **EPCList**
  - **EPC**: urn:epc:id:sscc:0614141.000000001
  - **EPC**: urn:epc:id:sscc:0614141.000000002
- **ReadPoint**: RP-5
- **Action**: OBSERVE
- **BizStep**: receiving
- **extention**: PedigreeEnvelope

---

**Case EPCs**

- **Receive Cases from Manufacturer**
  - **EPCIS Transaction Event**
    - **EventTime**: 2006-06-25T08:21:00Z
    - **EPCList**
      - **EPC**: urn:epc:id:sscc:0614141.000000001
      - **EPC**: urn:epc:id:sscc:0614141.000000002
    - **ReadPoint**: RP-5
    - **Action**: OBSERVE
    - **BizStep**: receiving
    - **extention**: PedigreeEnvelope

---

**Person in charge**: Wholesaler

**Using System**

- Wholesaler Capturing App. & EPCIS
Pedigree Scenario with EPCIS Event

◆ Step 11: **Wholesaler Publish Transaction Event (receiving)**
  - EPCIS Continuous query
    - Event Based Query
    - Used to **Obtain Transaction Event (receiving)** is made in Step 10
    - Sender Matching
  - Person in charge: Wholesaler
  - Using System
    - Wholesaler EPCIS

![Diagram showing data exchange between Manufacturer and Wholesaler]
Pedigree Scenario with EPCIS Event

Step 12: **Customer Buy the Genuine Product**
- Customer search Pedigree Data about Product
- Customer buy a genuine product
- **Update Pedigree Data**
  - Sold item serial number: **Kill State**

**<Product Info>**
1. Product Name: ABC Alcohol
3. Manufacturer: Kim Gil dong

**<Distribution Info>**
4. Kyung Buk A Alcohol Manufacturer (2009. 4. 1, Kim Gil Dong)
5. Kyung Nam KimHae Alcohol Wholesaler (2009. 4. 5, Lee Jin Young)
6. Busan Seomyun Alcohol Retailer (2009. 4. 7, Park Ba Ro)
Pedigree System - Design

◆ Architecture with EPCIS

→ Capturing Application
  ▪ EPCIS Event Create
  ▪ Transaction Event include Pedigree
  ▪ Pedigree Create by Pedigree Lib.

→ EPCIS
  ▪ EPCIS Event & Pedigree Store

→ Accessing Application
  ▪ EPCIS Event & Pedigree Query

→ Partner’s EPCIS
  Send Query Result
  ▪ EPCIS Transaction Event
  ▪ Include Partner’s Pedigree
## Pedigree System - Design

### Pedigree Library API

<table>
<thead>
<tr>
<th>API Name</th>
<th>Input Param Name</th>
<th>Return Value</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreateInitialPedigree</td>
<td>productName, manufacturerName, productCode, expirationDate, quantity, itemserialNumber…</td>
<td>InitialPedigree</td>
<td>Create InitialPedigree Element</td>
</tr>
<tr>
<td>CreateShippedPedigree</td>
<td>sender, receiver, senderAddress, receiverAddress, productInfo, itemInfo…</td>
<td>ShippedPedigree</td>
<td>Create ShippedPedigree Element</td>
</tr>
<tr>
<td>CreateReceivedPedigree</td>
<td>sender, receiver, senderAddress, receiverAddress, productInfo, itemInfo,…</td>
<td>ReceivedPedigree</td>
<td>Create ReceivedPedigree Element</td>
</tr>
<tr>
<td>CreateSignature</td>
<td>certificateFile, Password</td>
<td>Signature</td>
<td>Create Signature Element</td>
</tr>
<tr>
<td>CreatePedigree</td>
<td>InitialPedigree, ShippedPedigree, ReceivedPedigree, Signature</td>
<td>Pedigree</td>
<td>Create Pedigree Element</td>
</tr>
<tr>
<td>CreatePedigreeEnvelope</td>
<td>Container, Pedigree</td>
<td>PedigreeEnvelope</td>
<td>Create PedigreeEnvelope Element</td>
</tr>
<tr>
<td>ReceivePartnerPedigree</td>
<td>QueryResults</td>
<td>PartnerPedigree</td>
<td>Receive Partner’s Pedigree from Continuous QueryResults</td>
</tr>
</tbody>
</table>
Extended EPCIS Architecture for Pedigree

- Web Service Layer
- QueryService Layer
- Query Processing Layer
- Repository Layer
- Capture Service Layer

Data Flow:
- Data → Query

Components:
- EPCIS Web Server
- EPCIS Service Component
- EPCIS Rmi Server
- EPCIS Query Callback Component
- EPCIS Query Control Component
- Continuous(Subscribe) Query Component
- One Time(poll) Query Component

Data Sources:
- RFID Data
- RTLS Data
- Sensor Data
- Pedigree Data
- Master Data

Application Components:
- Capturing Application
- Capturing Application

Pedigree System - Design
Pedigree System - Design

◆ EPCIS Modified Class
  ➔ EPCISQueryControlImpl
    ▪ Poll Method Modify
  ➔ OneTimeQueryImpl
    ▪ Poll Method Modify
  ➔ SimpleQueryProcessor
    ▪ QueryProcessor Method Modify
  ➔ DataSourceAccessImpl
    ▪ PedigreeDataQuery Method Add
  ➔ DataSourceAccessInterface
    ▪ PedigreeDataQuery Method Add
  ➔ RDBDataSourceAccess
    ▪ pedigreeQuery Method Add
    ▪ generatePedigree Method Add
    ▪ generateTransactionEvent Method Modify

< LIT EPCIS Manager 3.0 Class Diagram >
Pedigree System - Design

**ER Diagram for Pedigree**

- **Pedigree**
  - pedigreeID
  - signatureID
  - signatureSN

- **DocumentInfo**
  - documentInfoID
  - serialNumber
  - version

- **ShippedPedigree**
  - pedigreeID
  - documentInfoID
  - initialPedigreeID
  - transactionInfoID
  - transactionInfo

- **ReceivedPedigree**
  - pedigreeID
  - documentInfoID
  - prePedigreeID
  - receivingInfoID
  - signatureInfoID

- **InitialPedigree**
  - initialPedigreeID
  - serialNumber
  - productInfoID
  - itemInfo

- **ReceivedInfo**
  - receivingInfoID
  - itemInfo
  - dateReceived

- **ItemInfo**
  - ItemInfoID
  - item
  - expirationDate
  - quantity
  - ItemSerialNumberInfoID

- **ItemSerialNumberInfo**
  - ItemSerialNumberInfoID
  - itemSerialNumber
  - ID

- **TransactionInfo**
  - transactionInfoID
  - sender_businessAddressID
  - sender_contactInfoID
  - nonPlant_businessAddressID
  - transactionIdentifierID

- **BusinessAddress**
  - businessAddressID
  - businessName
  - street
  - city
  - state
  - postalCode
  - country

- **ContactInfo**
  - contactInfoID
  - name
  - title
  - telephone
  - email

- **TransactionID**
  - transactionIdentifierID
  - identifier
  - identifierType
  - transactionType
  - transactionIdentifier

- **SignerInfo**
  - signerInfoID
  - name
  - title
  - telephone
  - email
  - url
  - signatureDate
  - signatureMeaning

- **Signature**
  - signatureInfoID
  - canonicalizationMethod
  - signatureMethodAlgorithm
  - referenceURI
  - transformAlgorithm
  - digestMethodAlgorithm
  - digestValue
  - signatureValue
  - keyInfo
  - PSSKeyValueModulus
  - PSSKeyValueExponent
  - SS01IssueName
  - SS01SerialNumber
  - SS01Certificate
Pedigree System - Design

- EPCIS Pedigree Query Interface
  - EPCIS Query Control Interface ✗ Poll
  - Query Type
    - **SimpleEventQuery**
      - Input Parameter: QueryParams
      - QueryResult: TransactionEvent List with Pedigree
    - **PedigreeDataQuery**
      - Input Parameter: QueryParams
        - Sender, Recipient, Signer, ShippingDate, ReceivingDate, ExpirationDate, EPC, ProductName, SenderCity, RecipientCity
      - QueryResult: Pedigree List

< SimpleEventQuery Example>

<table>
<thead>
<tr>
<th>QueryName</th>
<th>SimpleEventQuery</th>
<th>QueryName</th>
<th>PedigreeDataQuery</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Match_EPC</td>
<td>urn:epc:gid-96:1.1.1</td>
<td>Sender</td>
</tr>
<tr>
<td>QueryParams</td>
<td>eventType</td>
<td>TransactionEvent</td>
<td>Acme Laboratories</td>
</tr>
<tr>
<td></td>
<td>BizStep</td>
<td>Shipping</td>
<td>Shipping Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2009-02-04</td>
</tr>
</tbody>
</table>

< PedigreeDataQuery Example>
Pedigree System - Implementation

◆ Environment

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Intel Core(TM) 2 Quad CPU Q6600 2.4GHz</td>
</tr>
<tr>
<td>RAM</td>
<td>2GB</td>
</tr>
<tr>
<td>Operating System</td>
<td>Windows XP Pro Service Pack 3</td>
</tr>
<tr>
<td>Language</td>
<td>Java Development Kit(JDK) 1.6.02</td>
</tr>
<tr>
<td>Web Server</td>
<td>Sun Application Server 9.0</td>
</tr>
<tr>
<td>Database</td>
<td>Oracle 10g</td>
</tr>
</tbody>
</table>

◆ Implementation base on LIT EPCIS Manager 3.0
Pedigree System - Implementation

◆ EPCIS Accessing App. GUI
  ➞ SimpleEventQuery
    ▪ Transaction Event include Pedigree Data
  ➞ PedigreeDataQuery
    ▪ Direct Pedigree Query without Association with EPCIS Event
Conclusion & Further Research

◆ Conclusion

➔ Customer Needs Authentic Data in RFID Application
➔ Store Pedigree Data in every distribution step
  ▪ Complete, Accurate and Secure Records in Each Distribution
  ▪ Add to EPCIS Transaction Event
➔ EPCIS User Easy to Extend EPCIS for Pedigree
  ▪ Propose Pedigree System
    ▪ EPCIS User uses Pedigree Library for Pedigree Creating
➔ Customer kill the item serial number when buying the genuine product
  ▪ Prevent sale of the counterfeit product of the sold product’s same serial number

◆ Further Research

➔ The pedigree distribution system can be developed for the every EPCIS
  ▪ The distribution networks will be more transparent