Model Driven Development of Secure XML Data Warehouses: A Case Study

Belén Vela
Rey Juan Carlos University. Madrid. Spain

Carlos Blanco  Carlos.Blanco@unican.es
University of Cantabria. Santander. Spain

Eduardo Fernández-Medina
University of Castilla-La Mancha. Spain

Esperanza Marcos
Rey Juan Carlos University. Madrid. Spain
1. Introduction

- **Importance of Information Security**
  - The *survival* of organizations depends on the correct management of information security and confidentiality
  - Security constraints should be defined from *early stages* of the development process

- **Data Warehouses (DWs)**
  - Manage enterprise information for the decision making process
    - *Sensitive information* which should be protected

- **The Web and XML Technology**
  - Great amount of data available on the Web
    - DWs are considering the *Web* as data source
    - And *XML* to interchange data and metadata
1. Introduction

- **Model Driven Architecture (MDA)**
  - Is a *model oriented* approach for software development
  - Is based on the separation between
    - the *specification* of the system functionality and
    - its *implementation* by using specific platforms
  - Defines:
    - Models at different abstraction levels
    - Transformations between models

![Diagram showing MDA levels](image)
1. Introduction

- **Our proposal:**
  - **Development of Secure XML DWs** by using
    - A methodological approach based on MDA
    - **Confidentiality** issues (related with read operations)
    - A concrete technology based on the Web and XML
Classification:
- Levels (SL)
- Roles (SR)
- Compartments (SC)

Security Rules:
- Information Security
- Authorization
- Auditing

Cubes
- Dimensions
- Bases

UML profile for DWs

Secure Conceptual Metamodel (PIM)

Access Control and Audit (ACA) model
- Classification: Levels (SL)
- Roles (SR)
- Compartments (SC)
Secure XML Logical Metamodel (PSM)

- XML Technology
- Mappings from conceptual models (PIM)

1. Security Configuration
   - Security Levels, Roles and Compartments used in the model
   - Definition of a user profile

2. Star Package
   - Structural aspects
     - Fact, Dimensions, Bases,…
   - Security constraints
     - Security rules associated with MD elements
Secure XML Logical Metamodel (PSM) Security Configuration
Secure XML Logical Metamodell (PSM)

- Star Package
  - Structural aspects
    - Fact
    - Dimensions
    - Bases
    - Attributes
    …
Secure XML Logical Metamodel (PSM)

Star Package
- Security constraints
  - Security rules associated with MD elements (fact, dimensions,...)
Contents

1. INTRODUCTION
2. MDD OF SECURE XML DWs
3. CASE STUDY
4. CONCLUSIONS
• Case Study
  ▪ DW which manages airport’s information about trips involving passengers, baggage, flights, dates and places

  ▪ This example shows a partial view of this model
    ▪ A fact “Trip”
    ▪ A dimension “Place” related with a base “Airport”
    ▪ A dimension “Passenger”

  ▪ Security configuration used in this example
    ▪ Levels of security
      ▪ Top Secret (TS)
      ▪ Secret (S)
      ▪ Confidential (C)
      ▪ Unclassified (U)
    ▪ Hierarchy of security roles
      ▪ User
        ▪ Airport Security
        ▪ Administration
        ▪ Passenger
• Conceptual Model (structural aspects)
- Conceptual Model (security constraints)

```xml
<<SFact>>
Trip
{SL=C}
+idTrip
+price
+purpose {SR=Airport Security}
+seat
+distance
+flightTime
+checkIn
+boarding

<<SDimension>>
Place
{SL=C}
+deptPlace
+arrivPlace

<<SDimension>>
Airport
{SL=U}
+airportCode

<<SDimension>>
Passenger
{SL=S}
+passengerCode
+name
+address
+fingerprint {SR=Airport Security}
+passportPhoto {SR=Airport Security}
+suspicious {SR=Airport Security}

<<SecurityRule>> SIAR_TripPurpose
+ownedSCObjects = {Trip}
+involvedClasses = {Passenger}
+CABExp = "Trip.purpose=='military'"
+CATHENSecInf = {SL=S; SR=Airport Security}
+CAELSESecInf = {SL=C}

<<AuthorizationRule>> AUR_Passenger
+ownedSCObjects = {Passenger}
+ExceptSign = +
+ExceptPrivilege = read
+CABExp = "UserProfile.name == Passenger.name"

<<UserProfile>>
UserProfile
+UserCode
+name
+securityLevel
+securityRoles
```
• Logical Model

Schema

```xml
<?xml version="1.0" encoding="utf-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="SecureMDXML" type="SecureMDXML_Type" />
  <xs:complexType name="SecureInformation_Type">
    <xs:sequence>
      <xs:element name="userCode" type="xs:integer" />
      <xs:element name="name" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

User Profile

```
<xs:complexType name="UserProfile_Type">
  <xs:sequence>
    <xs:element name="userCode" type="xs:integer" />
    <xs:element name="name" type="xs:string" />
    <xs:element name="SecureInformation" type="SecureInformation_Type" />
  </xs:sequence>
</xs:complexType>
```
• Logical Model

Fact

```xml
<xs:complexType name="StarPackage_Type">
  <xs:sequence>
    <xs:element name="SecureFactClasses">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Trip" maxOccurs="unbounded">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="idTrip" type="xs:integer" />
                <xs:element name="price" type="xs:integer" />
                <xs:element name="purpose" type="xs:string" />
                <xs:element name="seat" type="xs:string" />
                <xs:element name="distance" type="xs:string" />
                <xs:element name="flightTime" type="xs:string" />
                <xs:element name="checkIn" type="xs:string" />
                <xs:element name="boarding" type="xs:string" />
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element name="SecurityLevel" fixed="C" />
          <xs:element name="SecurityRole" type="xs:string" fixed="AirportSecurity" />
          <xs:element name="deptPlace" type="xs:string" />
          <xs:element name="arrvPlace" type="xs:string" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

3. Case Study
MDD of Secure XML DWs: A Case Study

3. Case Study

• Logical Model

Security Rule
Associated with the Fact “Trip”
Logical Model

Dimensions and Bases

3. Case Study
Logical Model

Security Rule
Associated with the Dimension “Passenger”
Contents

1. INTRODUCTION

2. MDD OF SECURE XML DWs

3. CASE STUDY

4. CONCLUSIONS
• Conclusions
  ▪ Development of Secure XML DWs
    ▪ Use of the Model Driven Approach (MDA)
    ▪ Secure Conceptual MD Model (PIM)
      ▪ Independent of the target logical MD model
    ▪ Secure Logical MD Model (PSM)
      ▪ Focused on the XML Technology
      ▪ Including structural and security issues
    ▪ Semi-automatic Mapping from conceptual models
4. Conclusions

• Further work
  ▪ To define the mappings by using QVT transformations
  ▪ To transform security constraints defined with OCL into XPATH
  ▪ To apply the proposal to new case studies

• Discussion
  ▪ Since DWs mainly deals with read operations we focus our security work on confidentiality
    ▪ ¿Should other security issues be modeled for DWs?
      ▪ Integrity,…
    ▪ ¿Does cloud warehouses include new security aspects that should be modeled?
Model Driven Development of Secure XML Data Warehouses: A Case Study

Thanks for your attention

Belen Vela
Rey Juan Carlos University. Madrid. Spain

Carlos Blanco Carlos.Blanco@unican.es
University of Cantabria. Santander. Spain

Eduardo Fernandez-Medina
University of Castilla-La Mancha. Spain

Esperanza Marcos
Rey Juan Carlos University. Madrid. Spain