

# POC - Interconnexion Systèmes Informatiques Douaniers UEMOA

## 1. Vue d'ensemble de l'Architecture

### Composants principaux

- **Système Douanier Pays A** (Prime Abord) - Port H2 8081
- **Système Douanier Pays B** (Destination) - Port H2 8082
- **Kit d'Interconnexion MuleSoft** - Port 8080
- **Système Commission UEMOA** - Port H2 8083
- **Simulateur BCEAO** - Port H2 8084

## 2. Structure des Données

### Base de Données Pays A (Prime Abord)

sql

-- Table des manifestes

```
CREATE TABLE manifestes (  
    id BIGINT PRIMARY KEY AUTO_INCREMENT,  
    numero_manifeste VARCHAR(50) UNIQUE,  
    transporteur VARCHAR(100),  
    port_embarquement VARCHAR(50),  
    port_debarquement VARCHAR(50),  
    date_arrivee DATE,  
    statut VARCHAR(20) DEFAULT 'EN_ATTENTE'  
);
```

-- Table des marchandises

```
CREATE TABLE marchandises (  
    id BIGINT PRIMARY KEY AUTO_INCREMENT,  
    manifeste_id BIGINT,  
    code_sh VARCHAR(10),  
    designation VARCHAR(200),  
    poids_brut DECIMAL(10,2),  
    nombre_colis INTEGER,  
    destinataire VARCHAR(100),  
    pays_destination VARCHAR(3),  
    FOREIGN KEY (manifeste_id) REFERENCES manifestes(id)  
);
```

-- Table des échanges (trigger simulation)

```
CREATE TABLE echanges_sortants (  
    id BIGINT PRIMARY KEY AUTO_INCREMENT,  
    type_operation VARCHAR(50),  
    payload CLOB,  
    destination VARCHAR(50),  
    date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    statut VARCHAR(20) DEFAULT 'PENDING'  
);
```

## Base de Données Pays B (Destination)

sql

-- Table des déclarations

```
CREATE TABLE declarations (  
    id BIGINT PRIMARY KEY AUTO_INCREMENT,  
    numero_declaration VARCHAR(50) UNIQUE,  
    manifeste_origine VARCHAR(50),  
    declarant VARCHAR(100),  
    pays_origine VARCHAR(3),  
    date_depot TIMESTAMP,  
    statut VARCHAR(20) DEFAULT 'DEPOSEE'  
);
```

-- Table des Liquidations

```
CREATE TABLE liquidations (  
    id BIGINT PRIMARY KEY AUTO_INCREMENT,  
    declaration_id BIGINT,  
    montant_droits DECIMAL(12,2),  
    montant_taxes DECIMAL(12,2),  
    montant_total DECIMAL(12,2),  
    date_liquidation TIMESTAMP,  
    FOREIGN KEY (declaration_id) REFERENCES declarations(id)  
);
```

-- Table des paiements

```
CREATE TABLE paiements (  
    id BIGINT PRIMARY KEY AUTO_INCREMENT,  
    liquidation_id BIGINT,  
    montant_paye DECIMAL(12,2),  
    mode_paiement VARCHAR(50),  
    reference_paiement VARCHAR(100),  
    date_paiement TIMESTAMP,  
    FOREIGN KEY (liquidation_id) REFERENCES liquidations(id)  
);
```

**Base de Données Commission UEMOA**

sql

-- Table de traçabilité

```
CREATE TABLE tracabilite_operations (  
  id BIGINT PRIMARY KEY AUTO_INCREMENT,  
  numero_operation VARCHAR(100),  
  pays_origine VARCHAR(3),  
  pays_destination VARCHAR(3),  
  type_operation VARCHAR(50),  
  donnees_echange CLOB,  
  date_operation TIMESTAMP,  
  statut VARCHAR(20)  
);
```

-- Table des statistiques

```
CREATE TABLE statistiques_flux (  
  id BIGINT PRIMARY KEY AUTO_INCREMENT,  
  periode VARCHAR(20),  
  pays_origine VARCHAR(3),  
  pays_destination VARCHAR(3),  
  nombre_operations INTEGER,  
  montant_total DECIMAL(15,2),  
  date_creation TIMESTAMP  
);
```

## 3. APIs MuleSoft - Kit d'Interconnexion

### 3.1 API de Réception (Pays A → Kit)

**Endpoint:** POST /api/v1/manifeste/transmission

json

```
{
  "numeroManifeste": "MAN2025001",
  "transporteur": "MAERSK",
  "portEmbarquement": "ROTTERDAM",
  "portDebarquement": "ABIDJAN",
  "dateArrivee": "2025-01-15",
  "marchandises": [
    {
      "codeSH": "8703.21.10",
      "designation": "Véhicule particulier",
      "poidsBrut": 1500.00,
      "nombreColis": 1,
      "destinataire": "IMPORT SARL",
      "paysDestination": "BFA"
    }
  ]
}
```

### 3.2 API de Transmission (Kit → Pays B)

**Endpoint:** `POST /api/v1/declaration/preparation`

### 3.3 API de Notification Paiement (Pays B → Kit)

**Endpoint:** `POST /api/v1/paiement/notification`

### 3.4 API de Mainlevée (Kit → Pays A)

**Endpoint:** `POST /api/v1/mainlevee/autorisation`

## 4. Flux de Données - Scénario Libre Pratique

### Étape 1: Prise en charge manifeste (Pays A)

1. Saisie manifeste dans système Pays A
2. Trigger automatique → transmission vers Kit MuleSoft
3. Kit route vers Pays B et Commission

### Étape 2: Déclaration (Pays B)

1. Réception données manifeste
2. Déclarant établit déclaration détaillée
3. Liquidation automatique des droits et taxes

### Étape 3: Paiement et mainlevée

1. Paiement effectué (simulation BCEAO)
2. Notification vers Kit MuleSoft
3. Kit autorise mainlevée Pays A
4. Archivage Commission UEMOA

## **5. Patterns MuleSoft à Implémenter**

### **5.1 Message Routing**

- Router par pays destination
- Load balancing si multiple instances

### **5.2 Data Transformation**

- XML ↔ JSON conversion
- Enrichissement données référentielles
- Validation business rules

### **5.3 Error Handling**

- Dead Letter Queue
- Retry policies
- Circuit breaker pattern

### **5.4 Security**

- API Key authentication
- OAuth 2.0 pour échanges sensibles
- Encryption/Decryption

## **6. Monitoring et Observabilité**

### **Métriques à suivre**

- Nombre de transactions par pays
- Temps de réponse moyen
- Taux d'erreur par endpoint
- Volume de données échangées

### **Dashboards**

- Tableau de bord Commission UEMOA
- Monitoring technique MuleSoft

- KPIs business par corridor

## **7. Configuration Anypoint Platform**

### **API Manager**

- Policies: Rate limiting, CORS, Security
- SLA Tiers par type d'utilisateur
- Analytics et monitoring

### **Runtime Manager**

- Deployment en CloudHub ou On-Premise
- Scaling automatique
- Health checks

### **Design Center**

- RAML specification des APIs
- Mocking services pour tests
- Documentation interactive