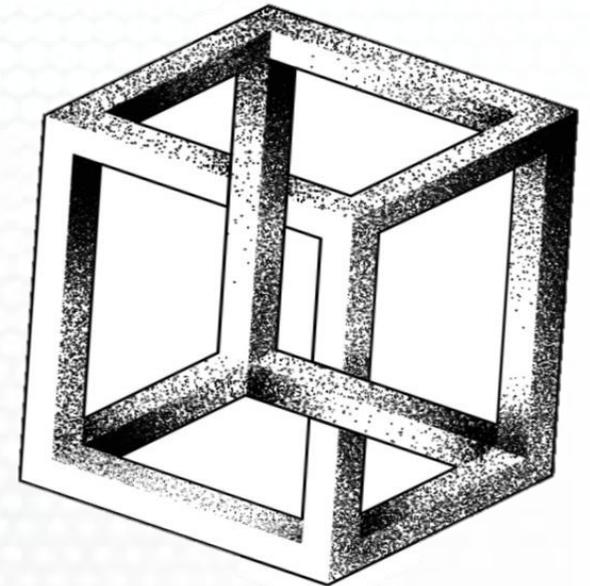


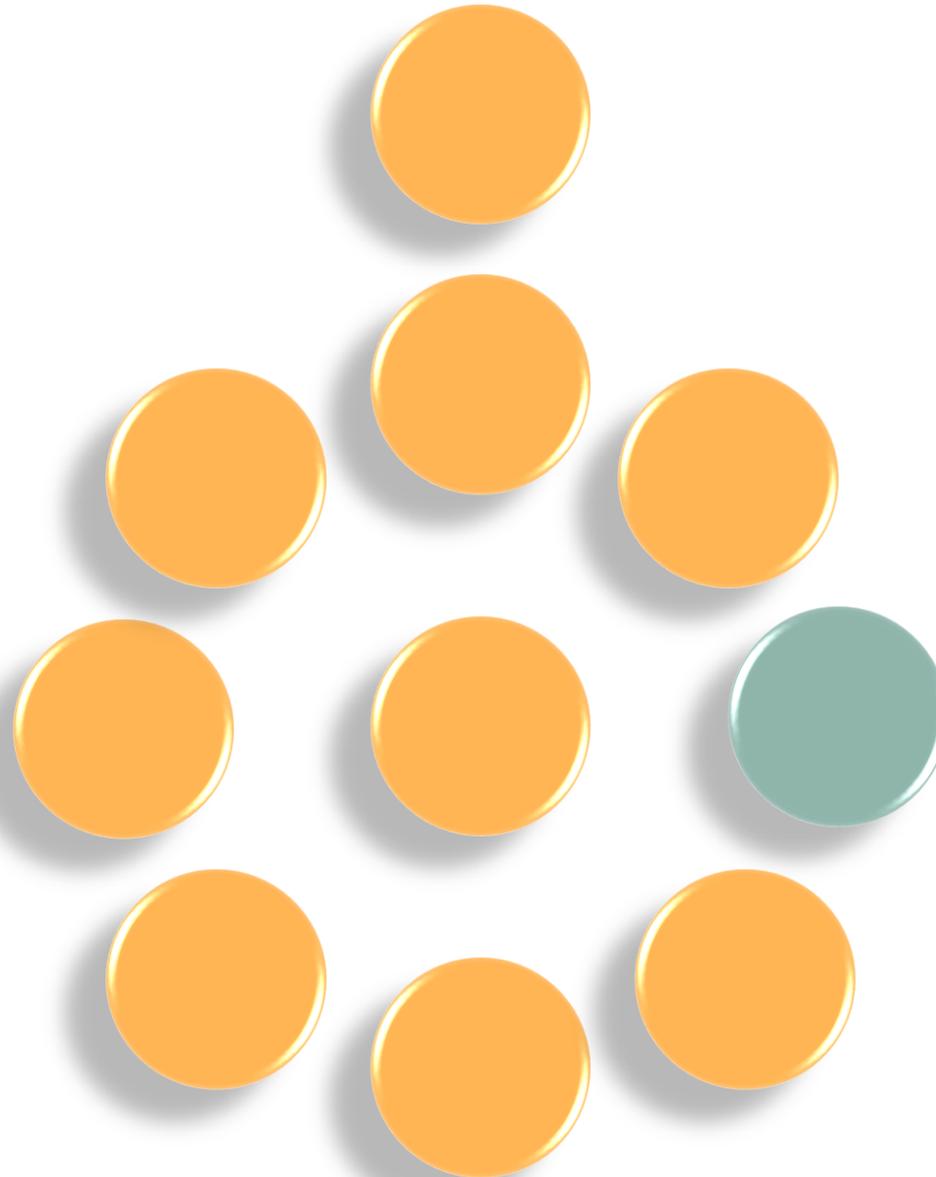
EA Day 2019

La donnée dans tous ses états

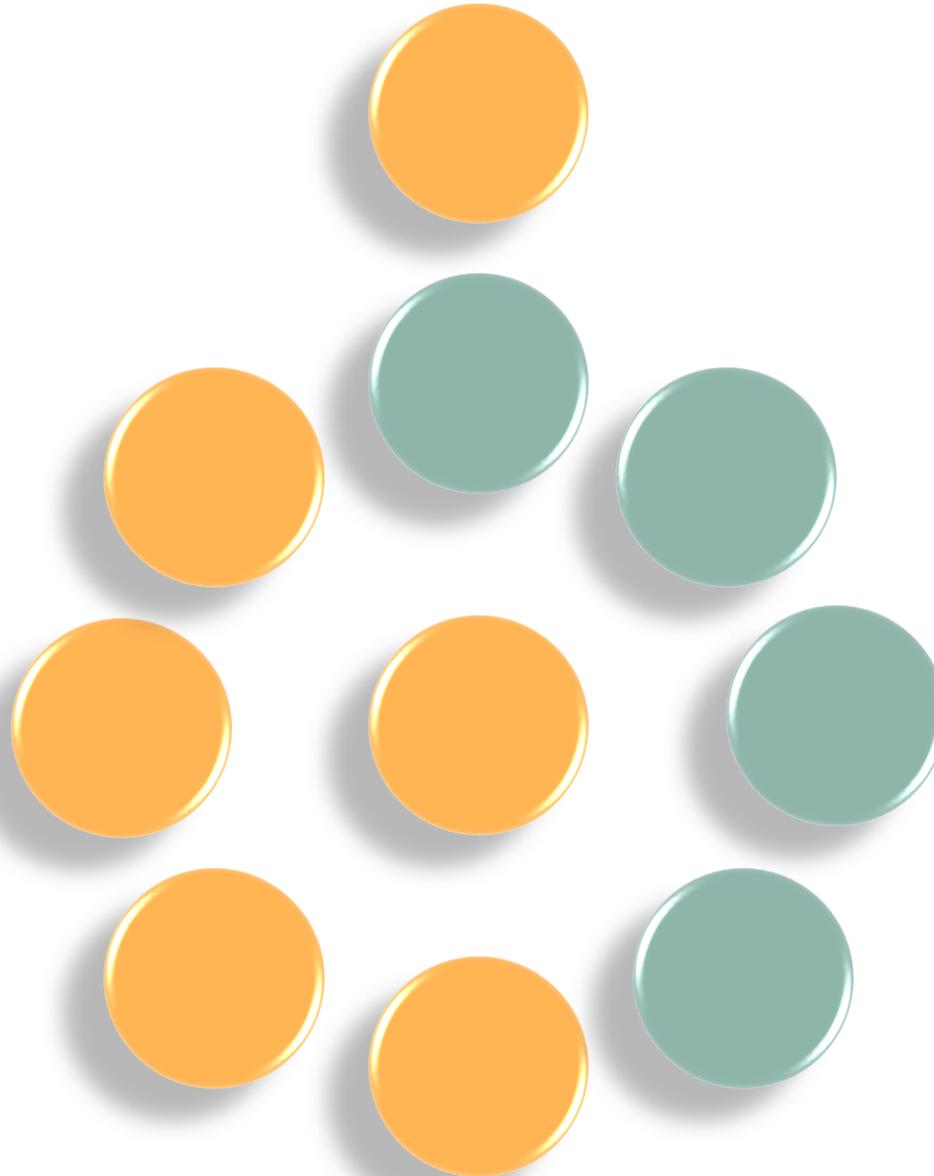
Maud Brevet

Etudier un système d'information  
sous **différents points de vue**





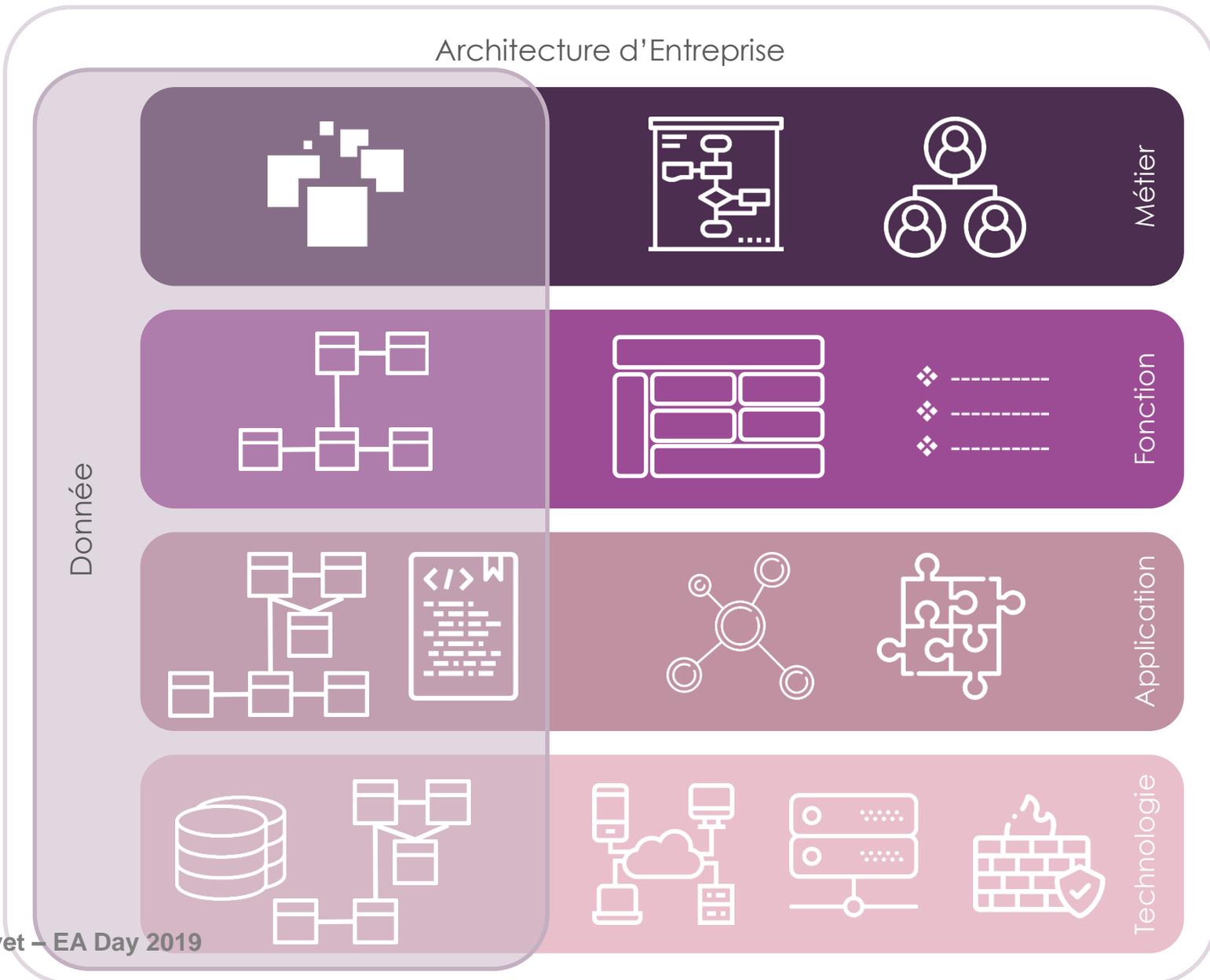
Phase C  
Data Architecture



La donnée

Depuis la vision  
jusqu'à la technologie

# La donnée est transverse



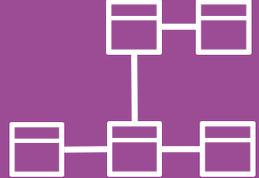
# Représentations de la donnée

## Architecture d'Entreprise



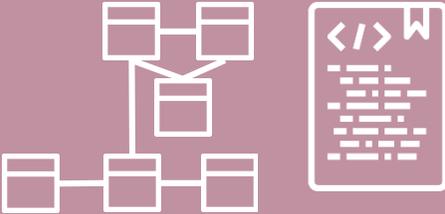
**Concept** métier

Métier



Modèle **conceptuel** de données

Fonction



Modèle **logique** de données

Modèle **canonique** de données

Application



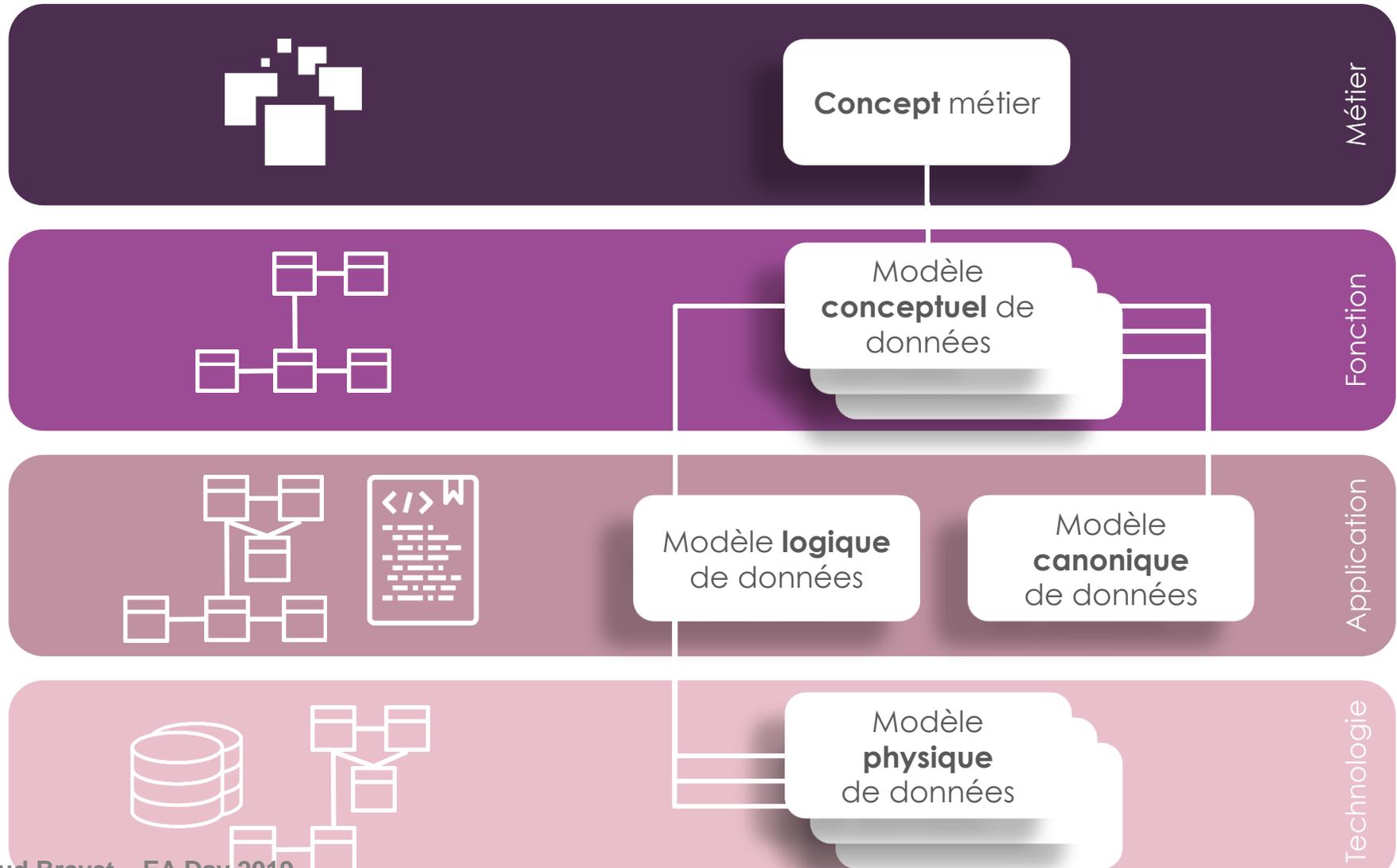
Modèle **physique** de données

Technologie



# Représentations de la donnée

## Architecture d'Entreprise



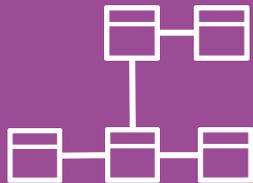
# Représentations de la donnée

## Architecture d'Entreprise



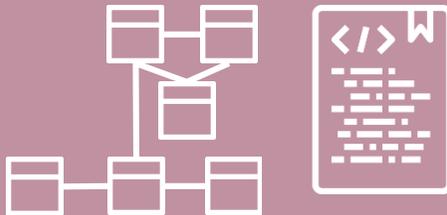
Concept métier

Métier



Modèle **conceptuel** de données

Fonction



Modèle **logique** de données

Modèle **canonique** de données

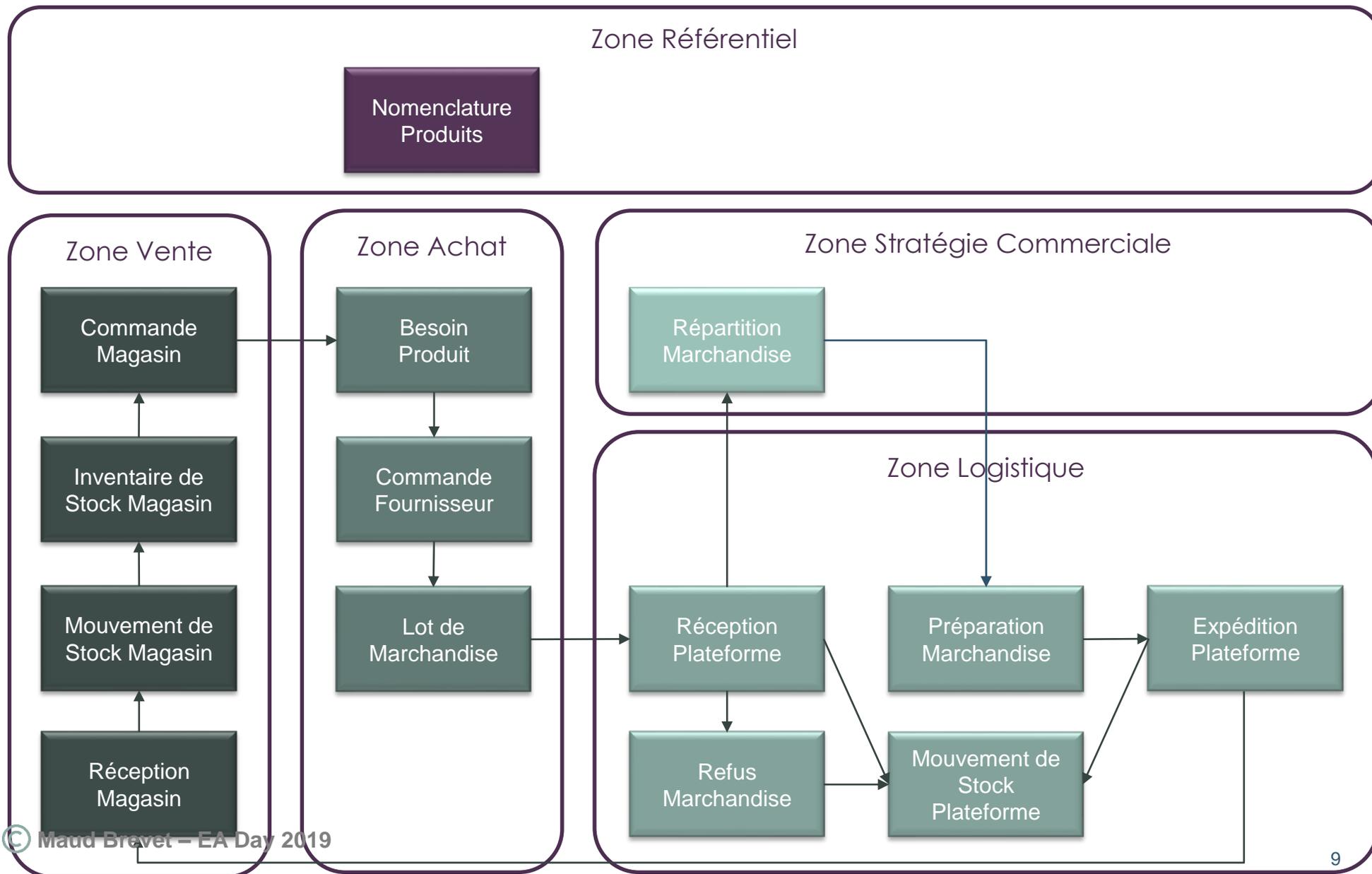
Application



Modèle **physique** de données

Technologie

# Vue d'ensemble des concepts métiers clés





Etablir un **glossaire d'entreprise**

→ Partager un **même langage** entre business & IT

Définir l'**ownership**

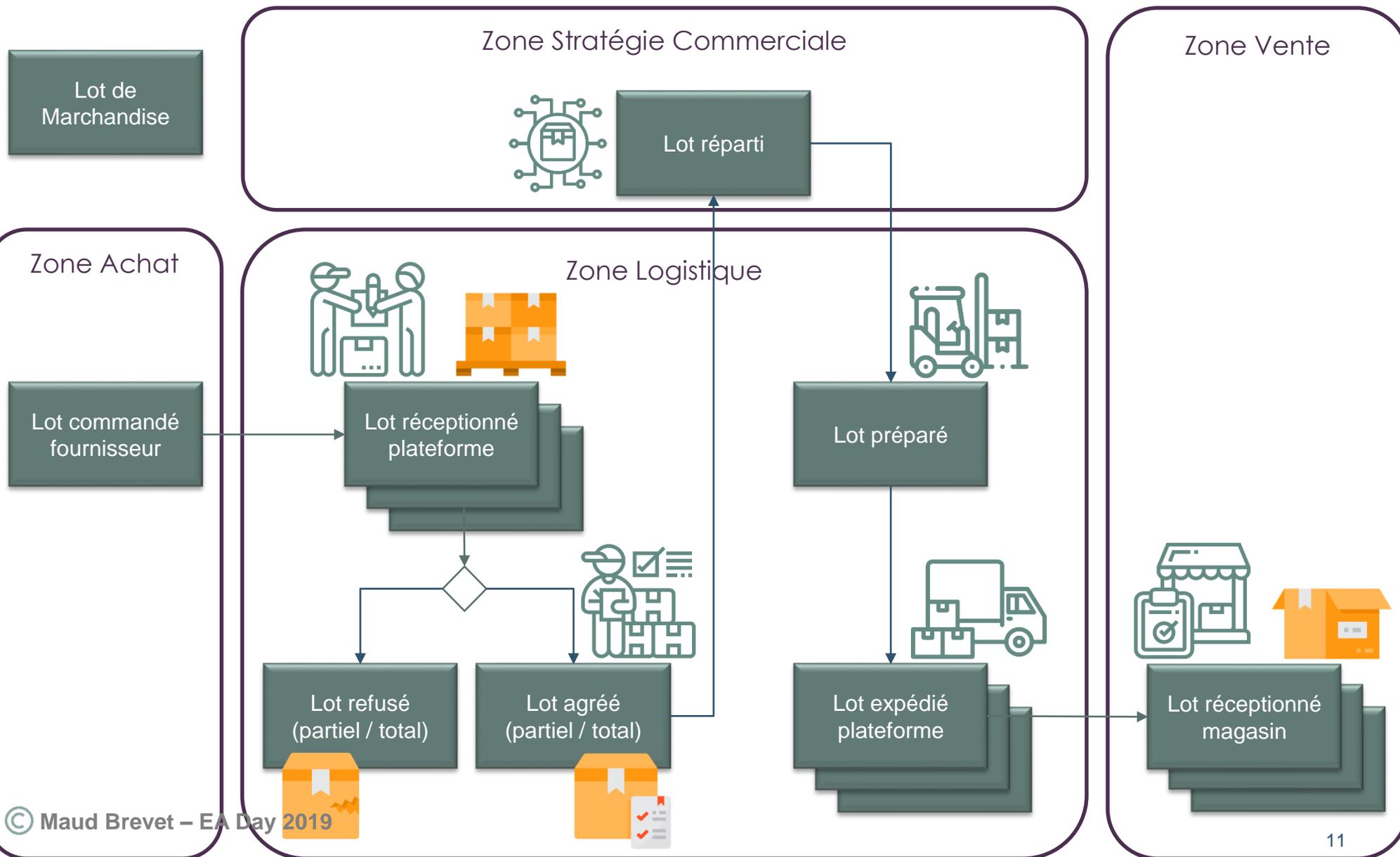
→ Définir quel domaine métier **donne naissance** à quel concept métier

Détecter les **synonymes**

→ La donnée porte des **noms différents** pour désigner la même chose

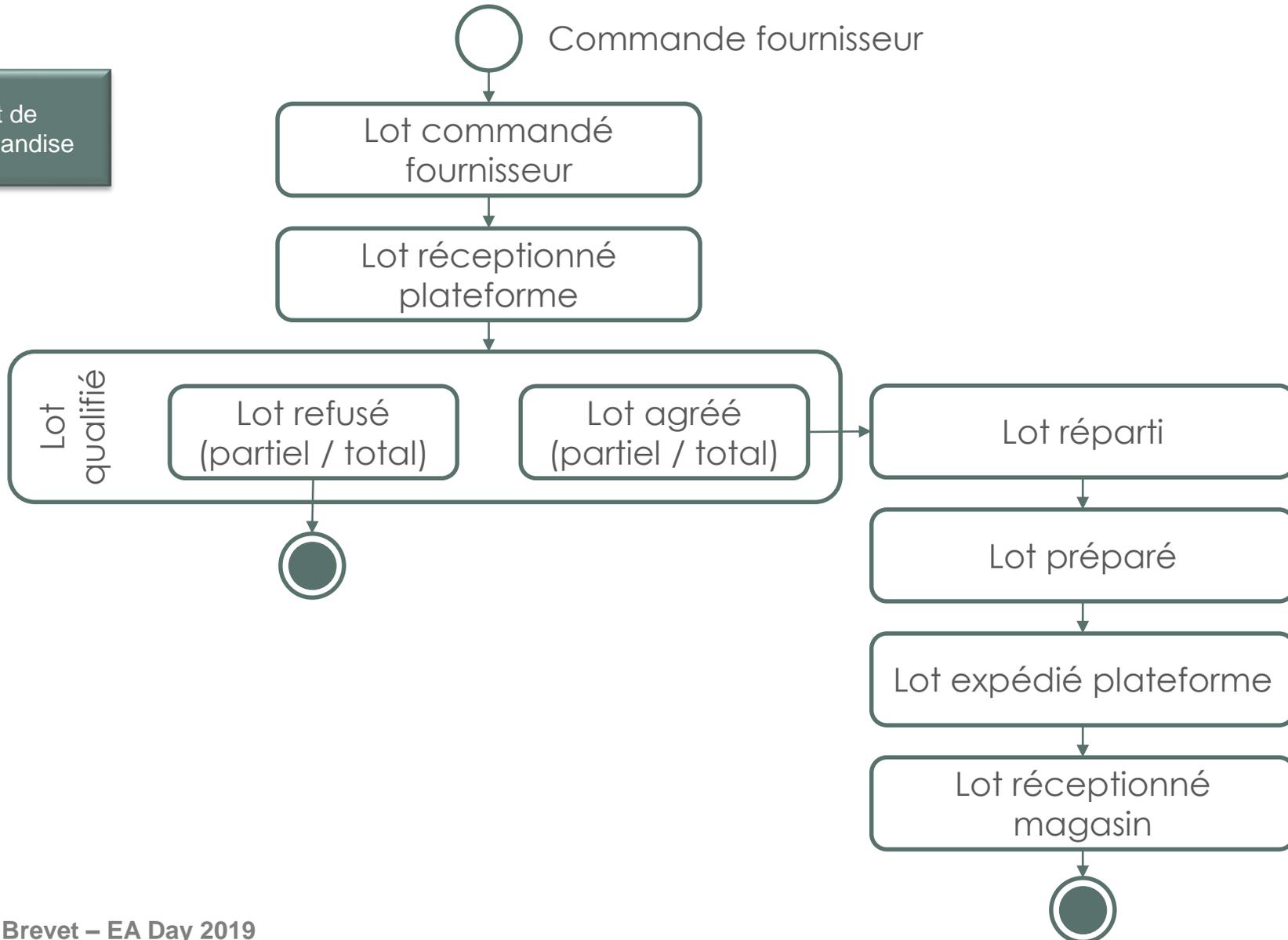
Détecter les **polysèmes**

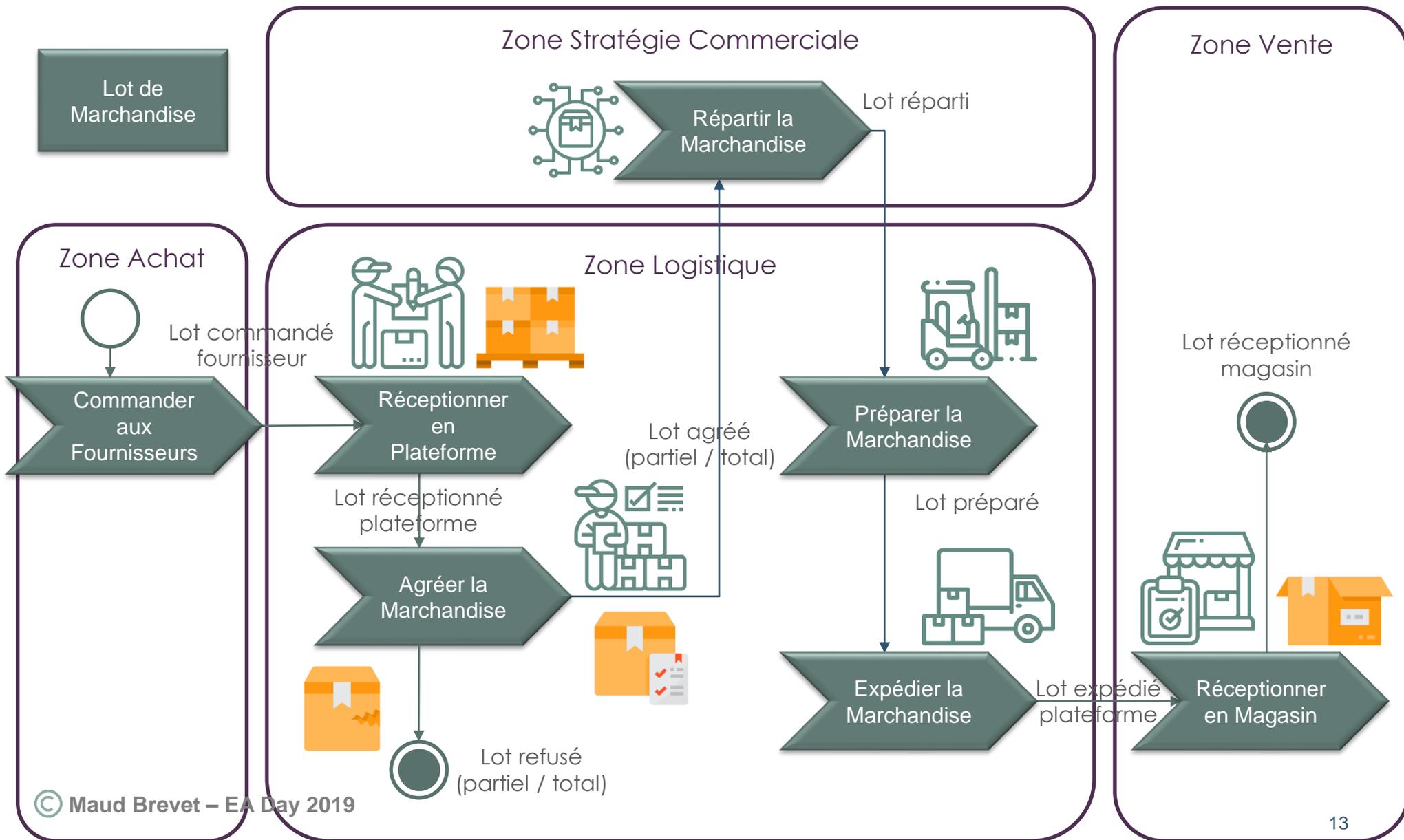
→ Des données portent le même nom mais représentent des **sens différents**





Lot de  
Marchandise





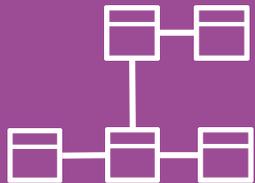
# Représentations de la donnée

## Architecture d'Entreprise



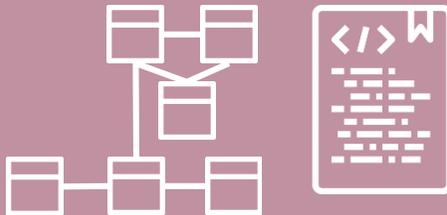
Concept métier

Métier



Modèle **conceptuel** de données

Fonction



Modèle **logique** de données

Modèle **canonique** de données

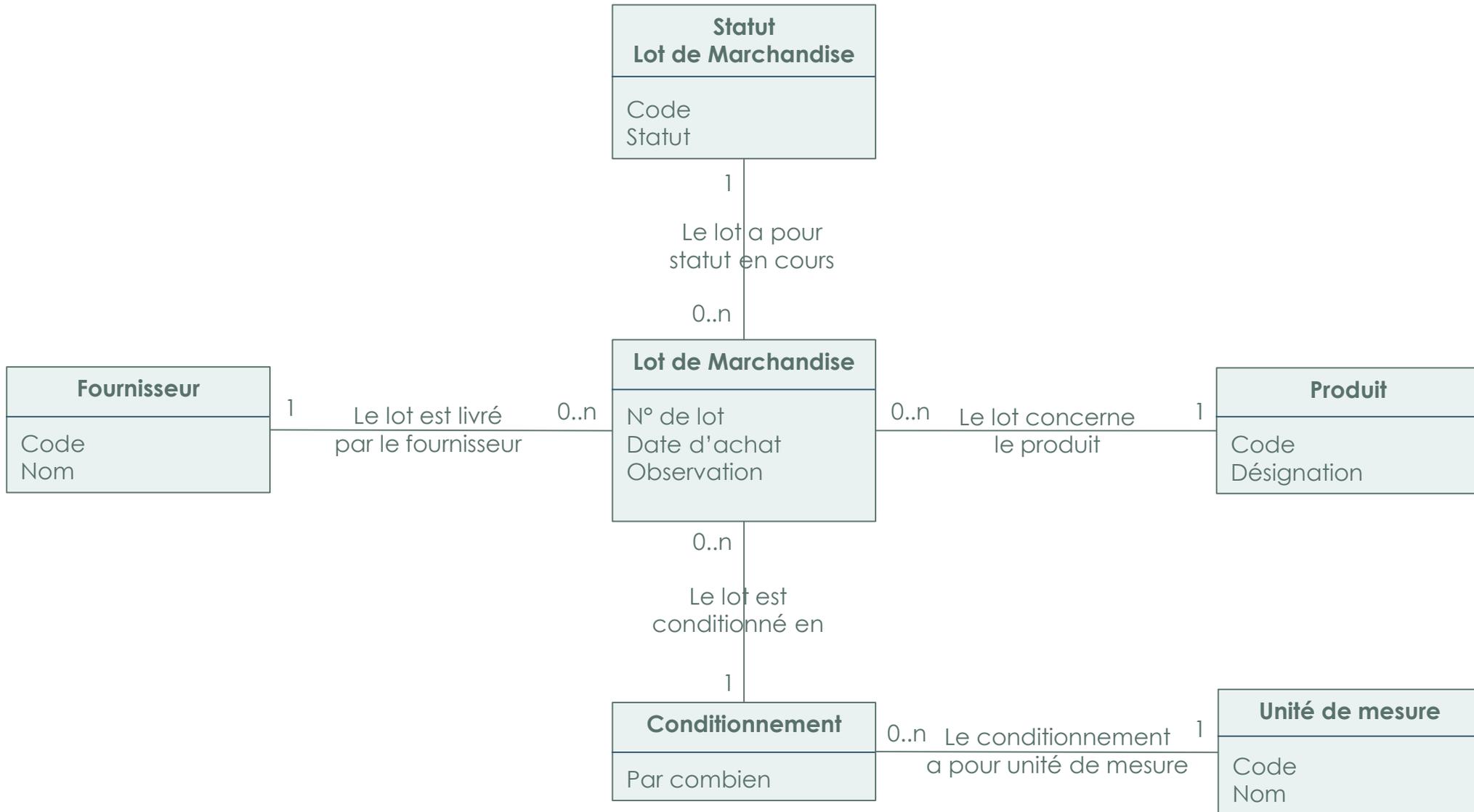
Application

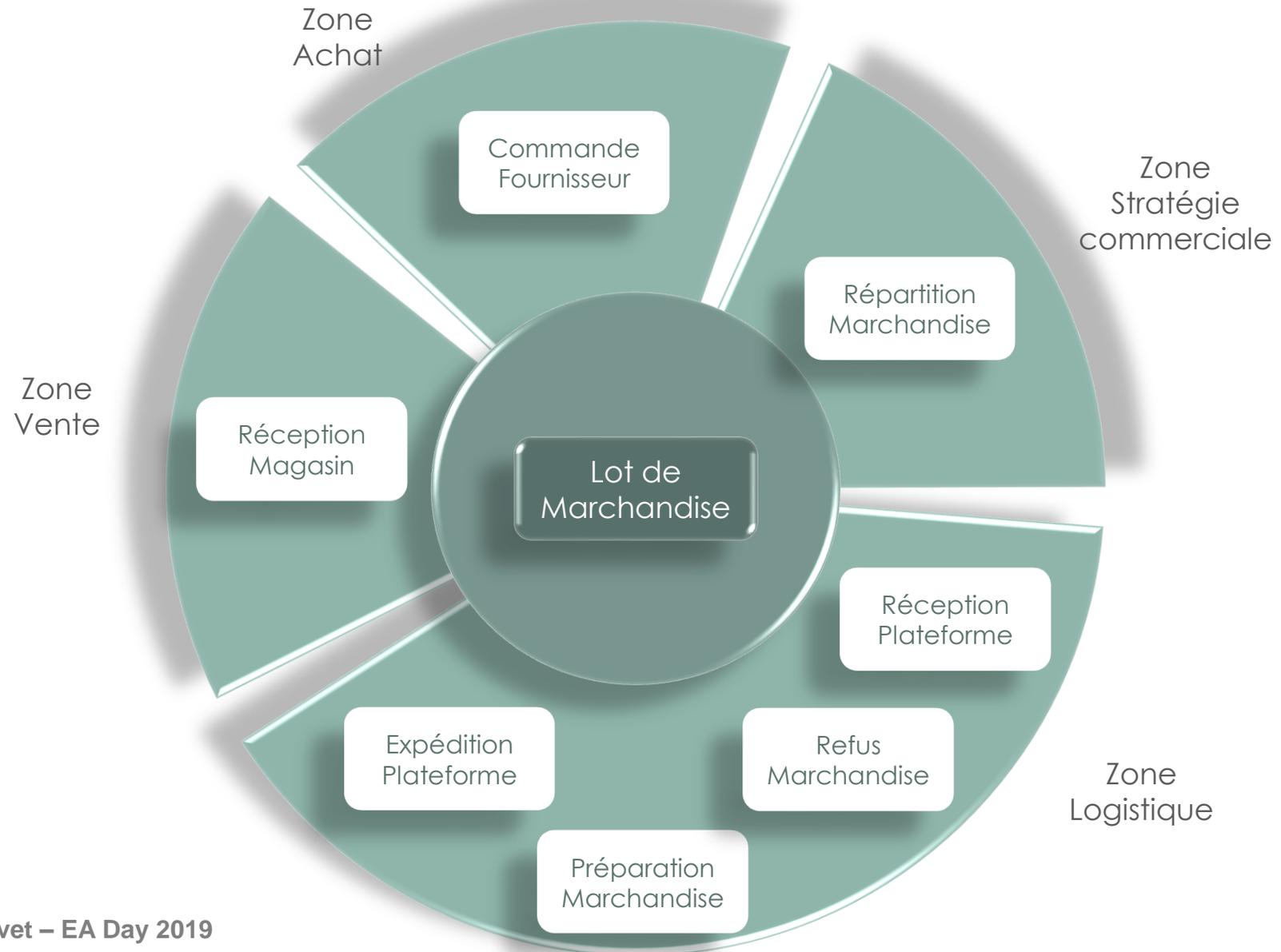


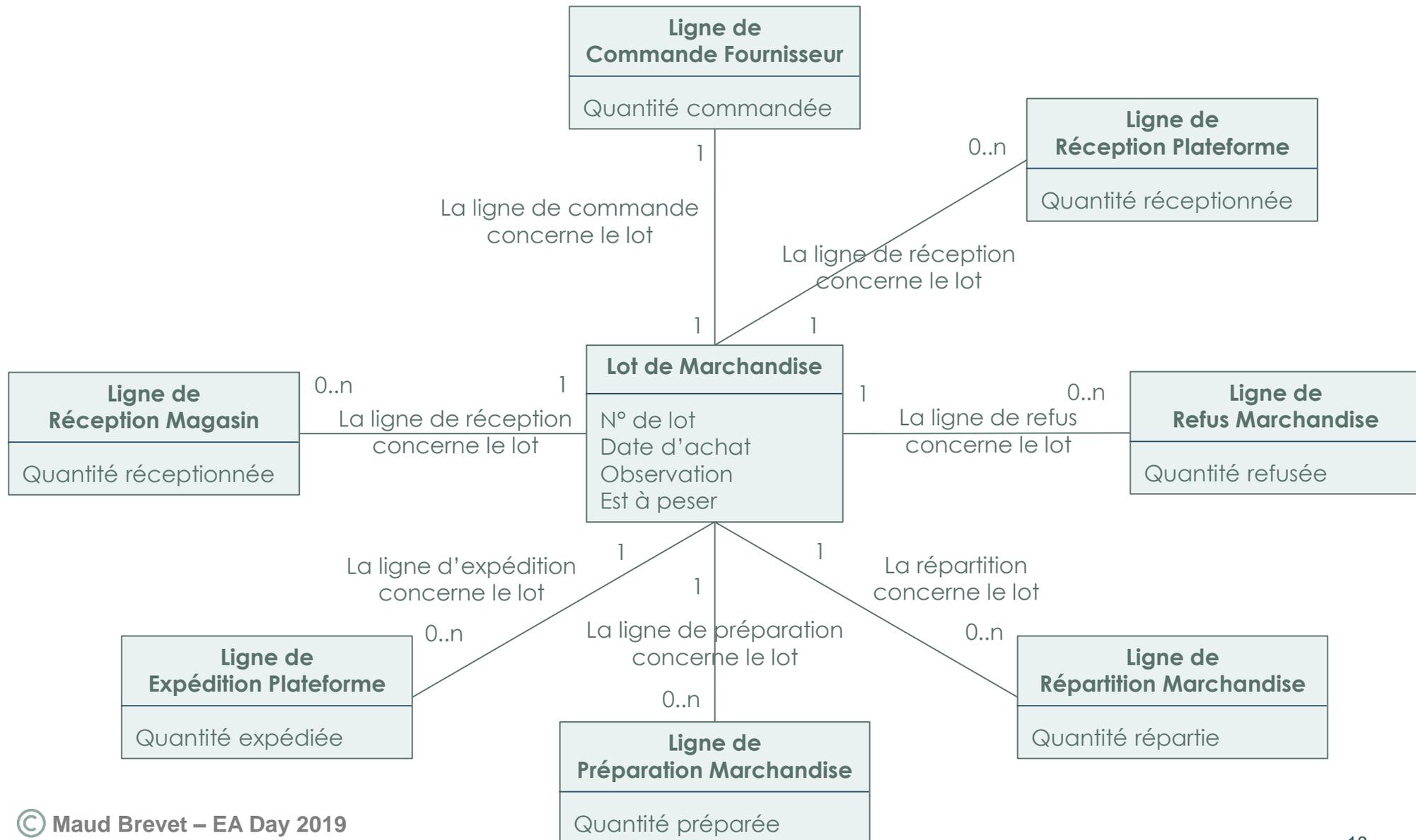
Modèle **physique** de données

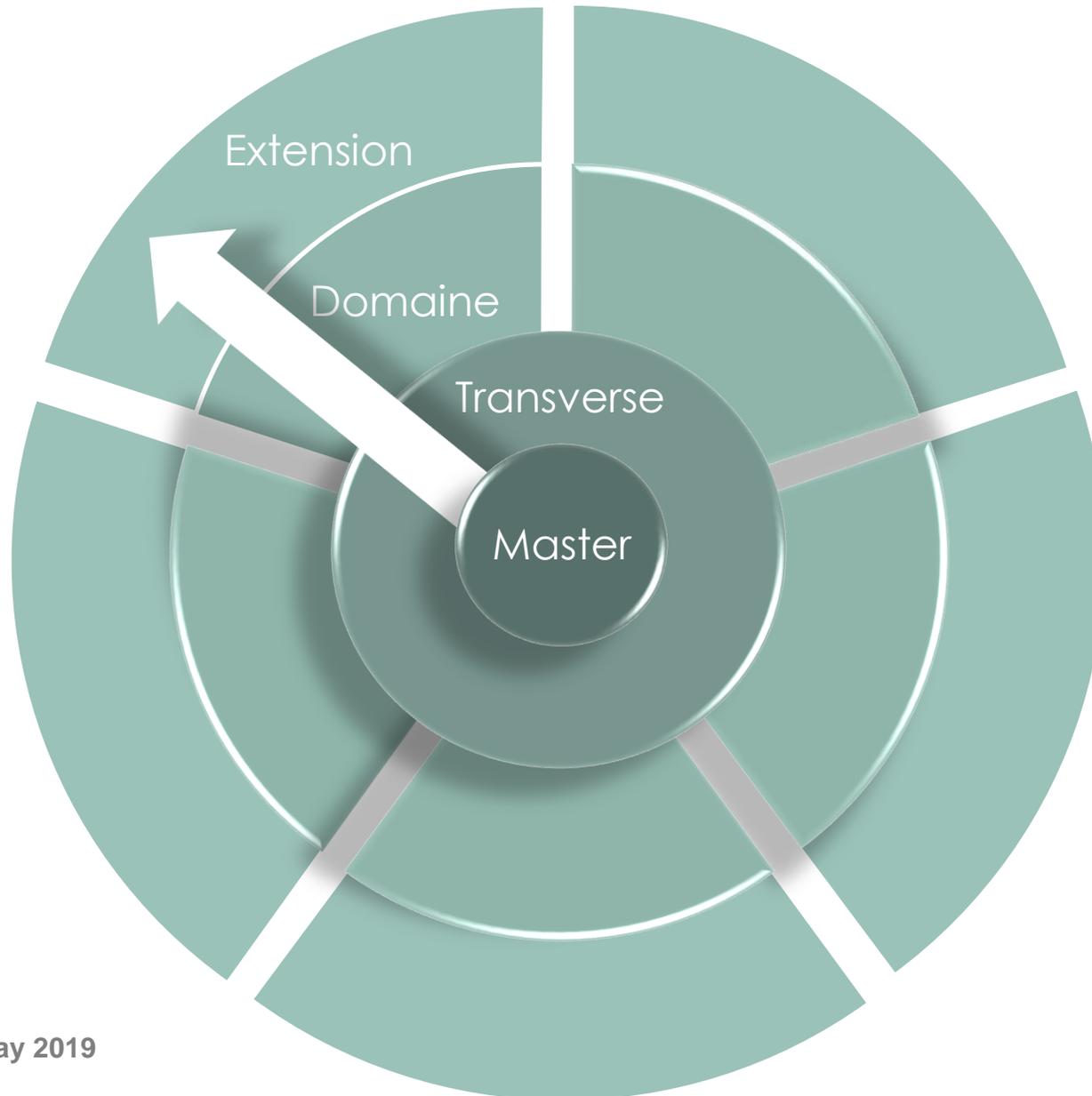
Technologie

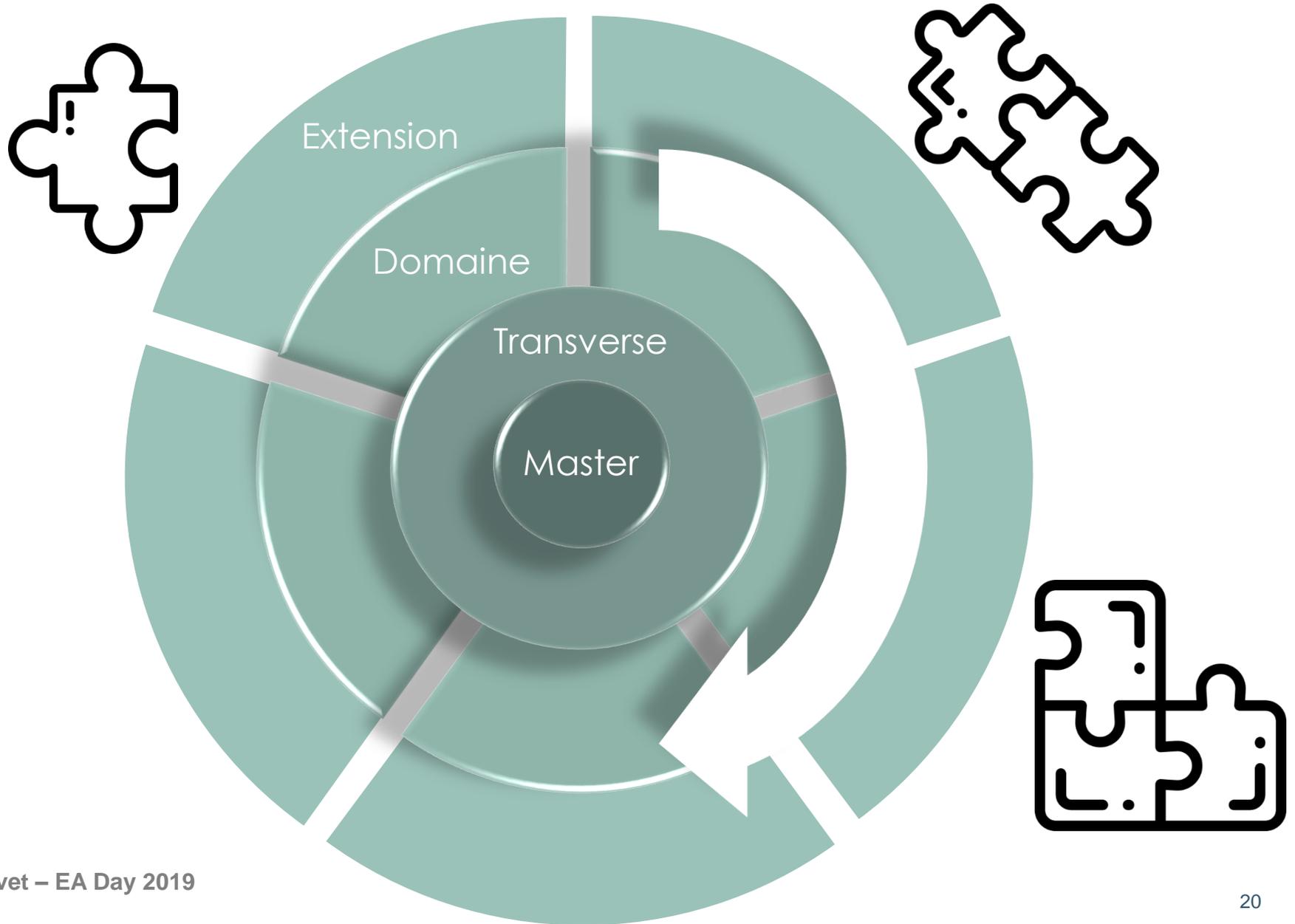
Accorder autant d'importance aux **liens entre les éléments** qu'aux éléments eux-mêmes

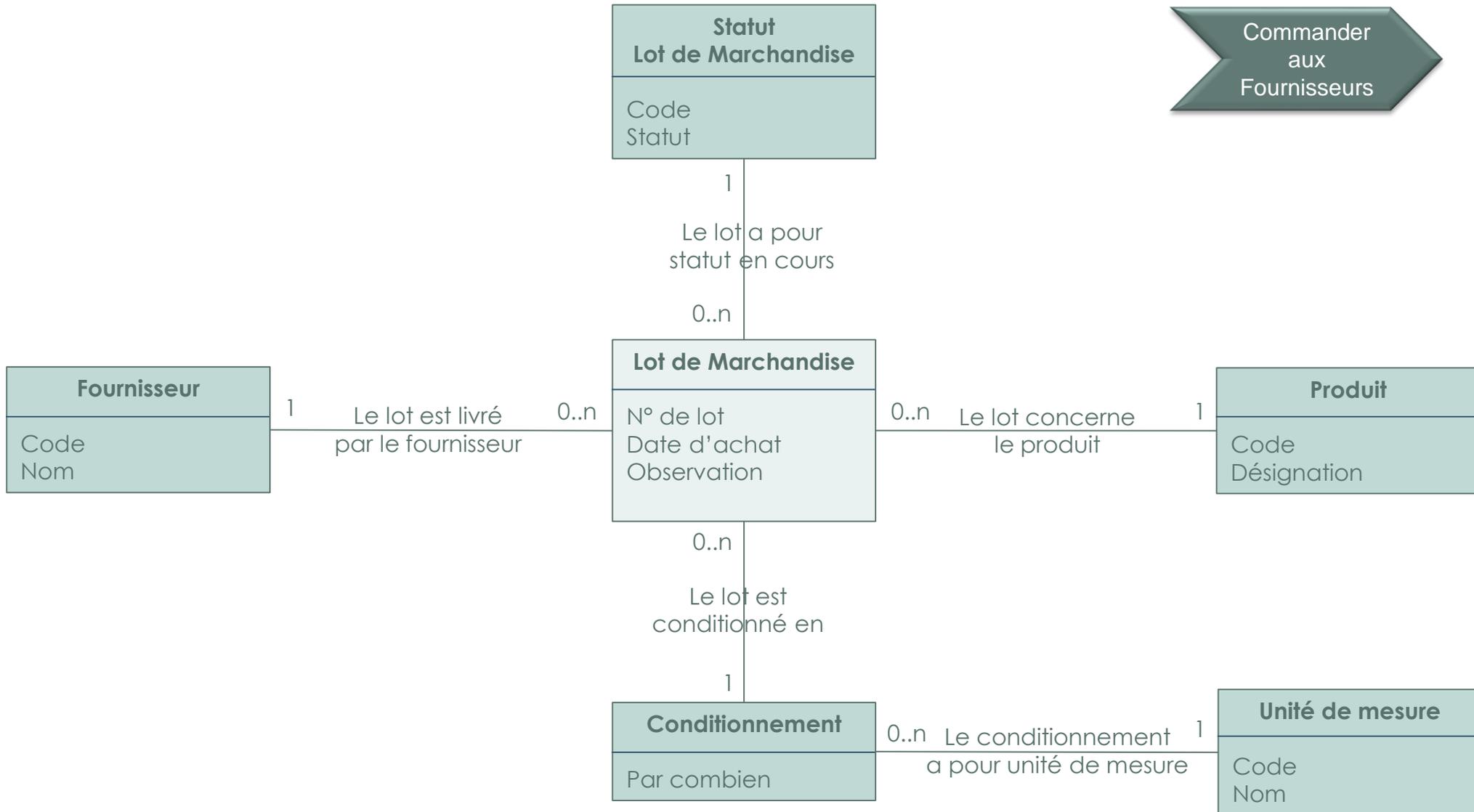


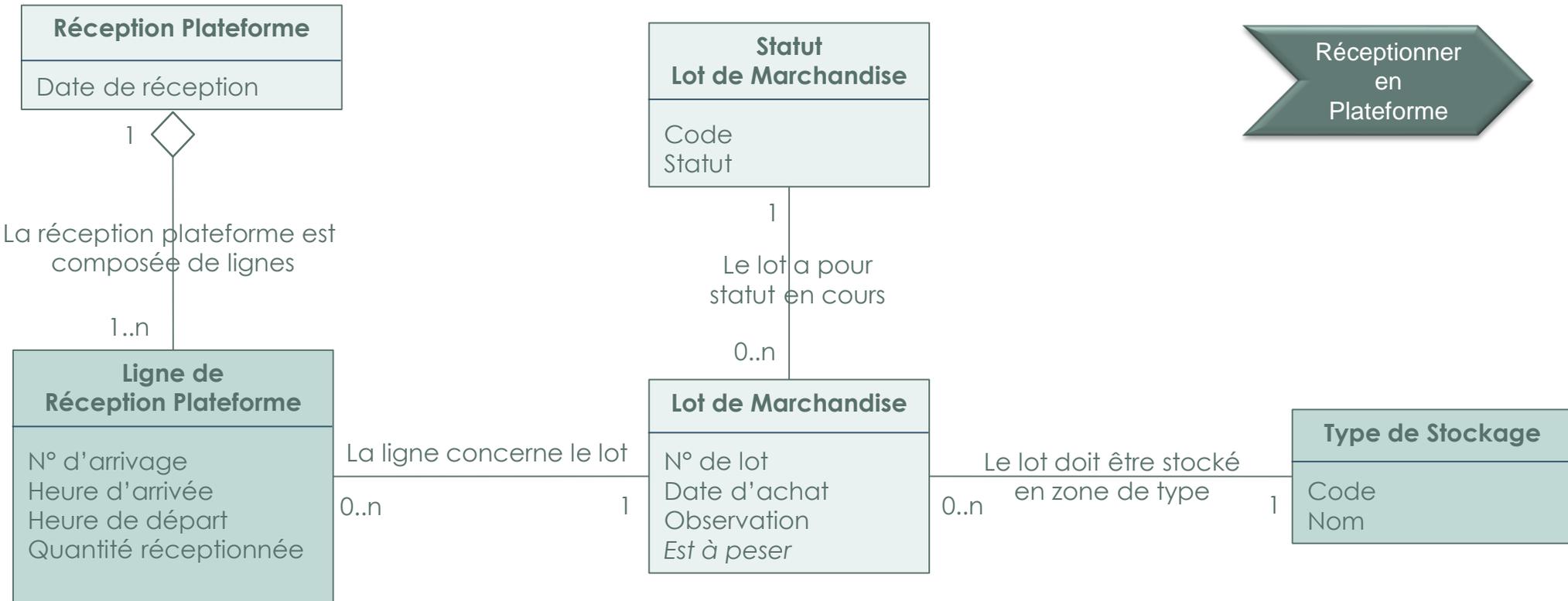


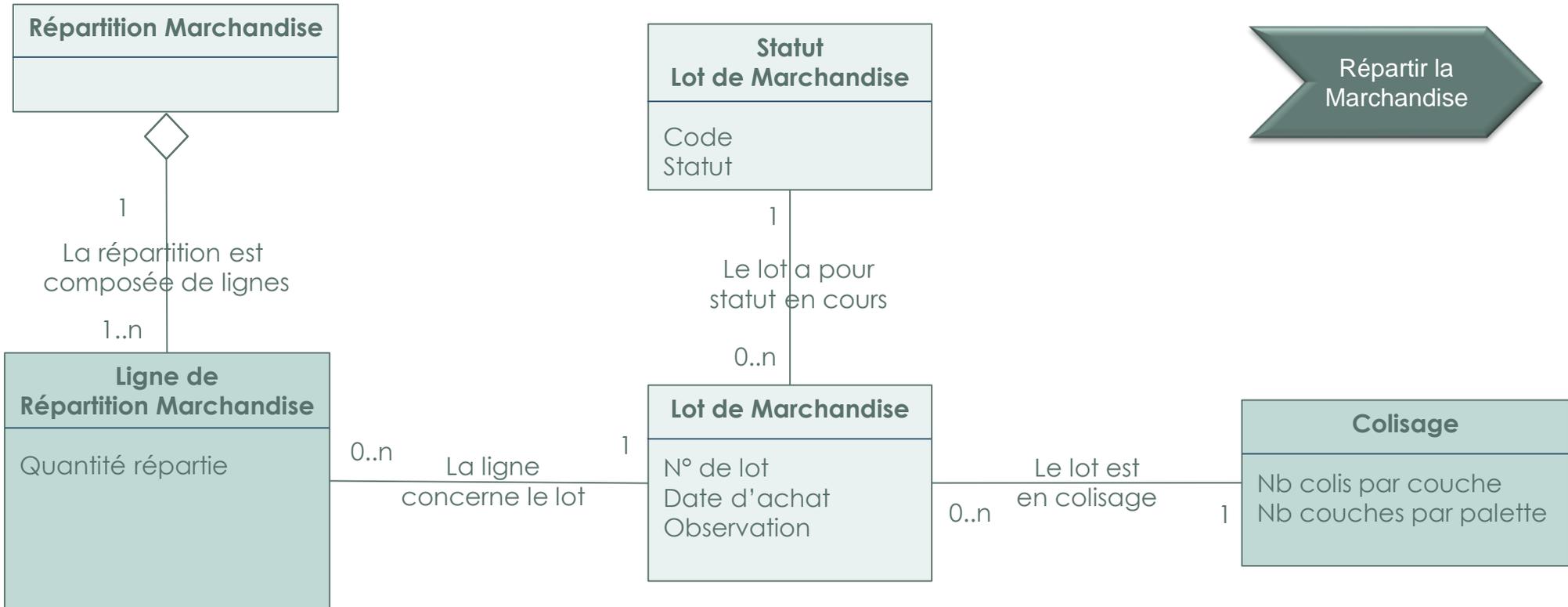












## Détailler les **propriétés**

→ Déterminer l'**ownership** des propriétés d'un concept métier tout au long de son cycle de vie

## Identifier les **liens sémantiques**

→ Mettre en évidence les **relations** entre les concepts métiers

## Préparer la vue **logique**

→ Préparer la **transformation** d'un modèle conceptuel **en un modèle logique**

## Préparer la vue **canonique**

→ Préparer la **transformation** de plusieurs modèles conceptuels **en un modèle canonique**

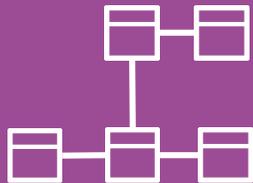
# Représentations de la donnée

## Architecture d'Entreprise



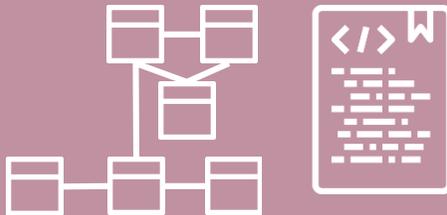
Concept métier

Métier



Modèle **conceptuel** de données

Fonction



Modèle **logique** de données

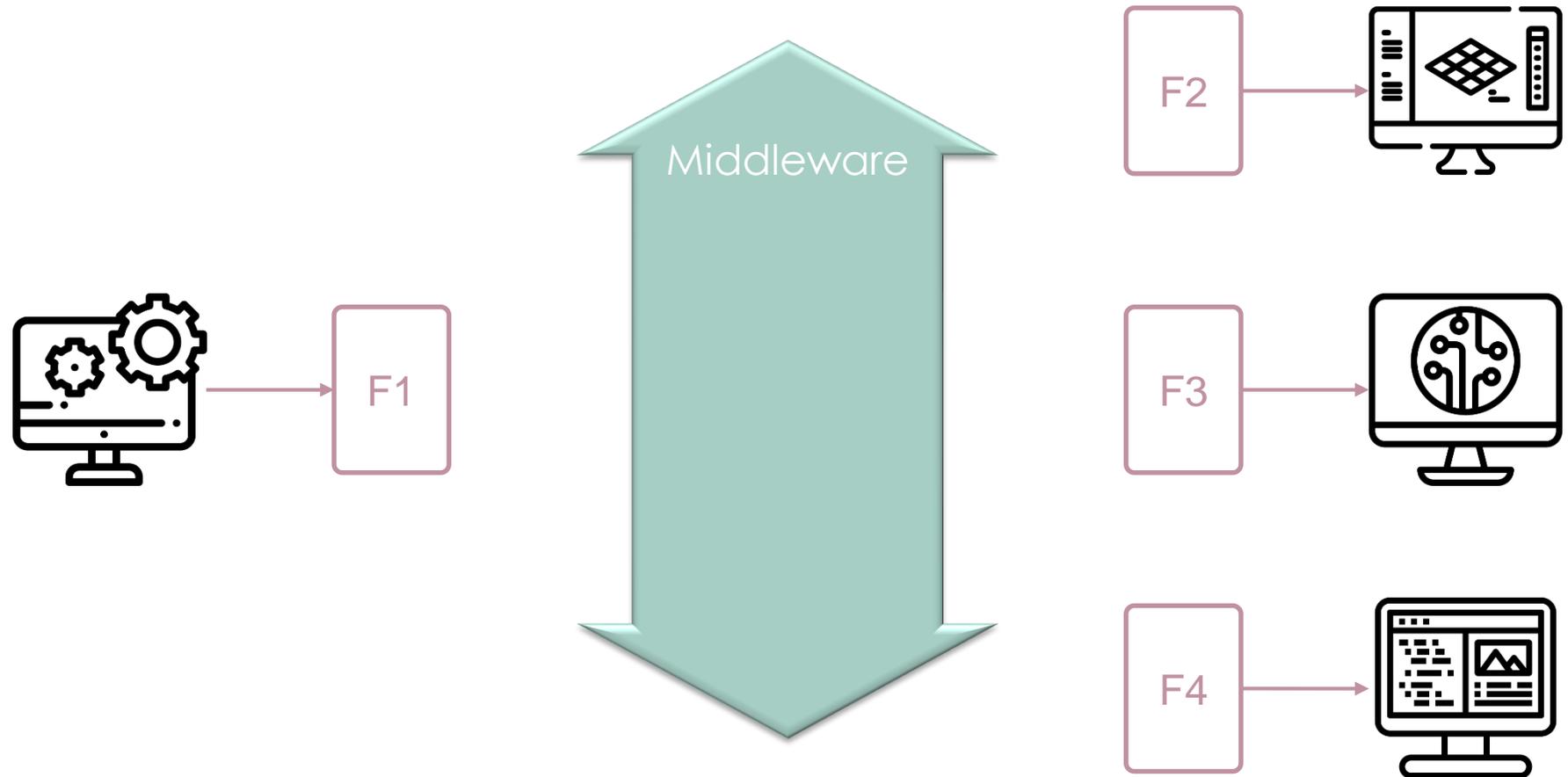
Modèle **canonique** de données

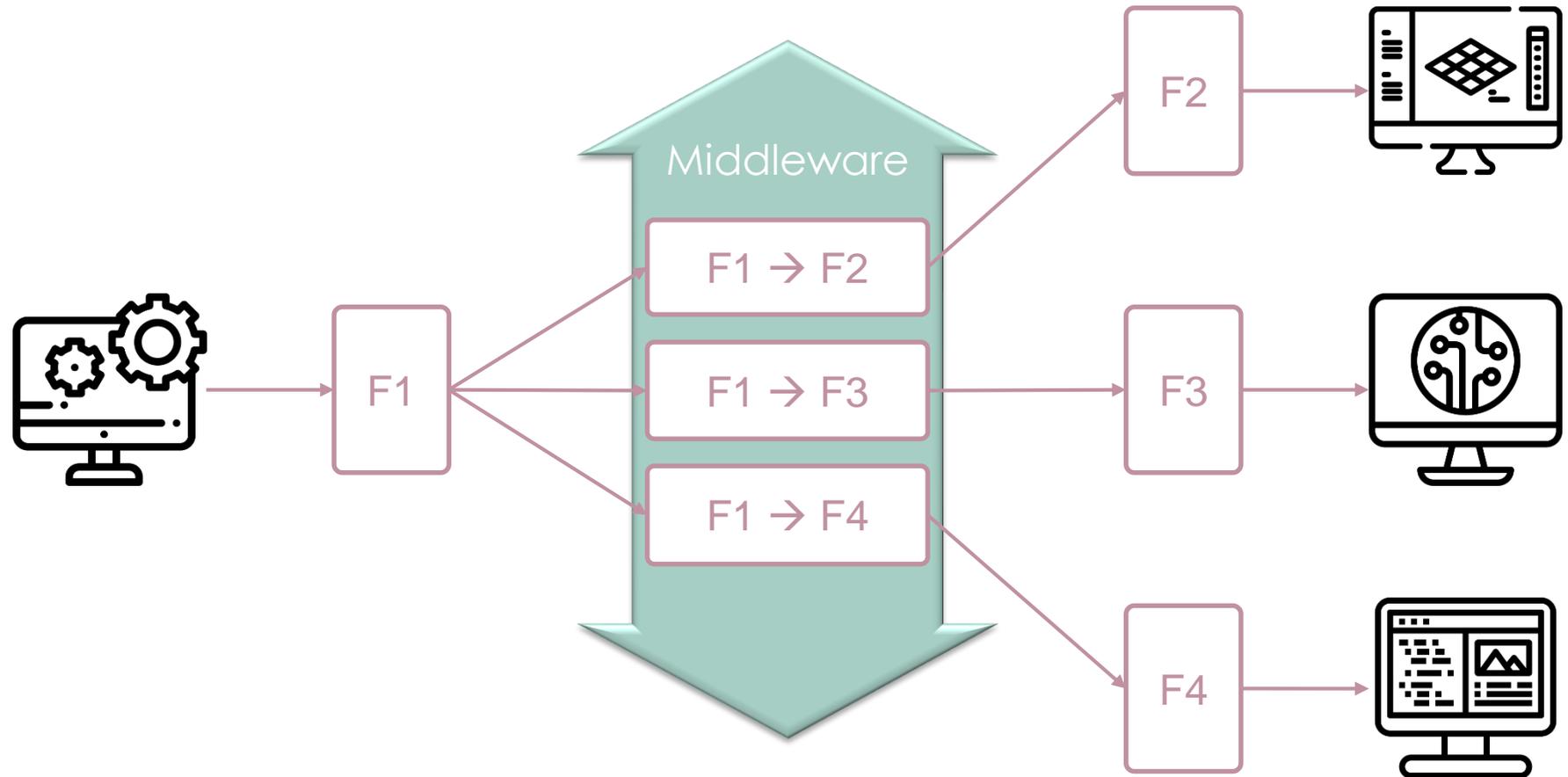
Application

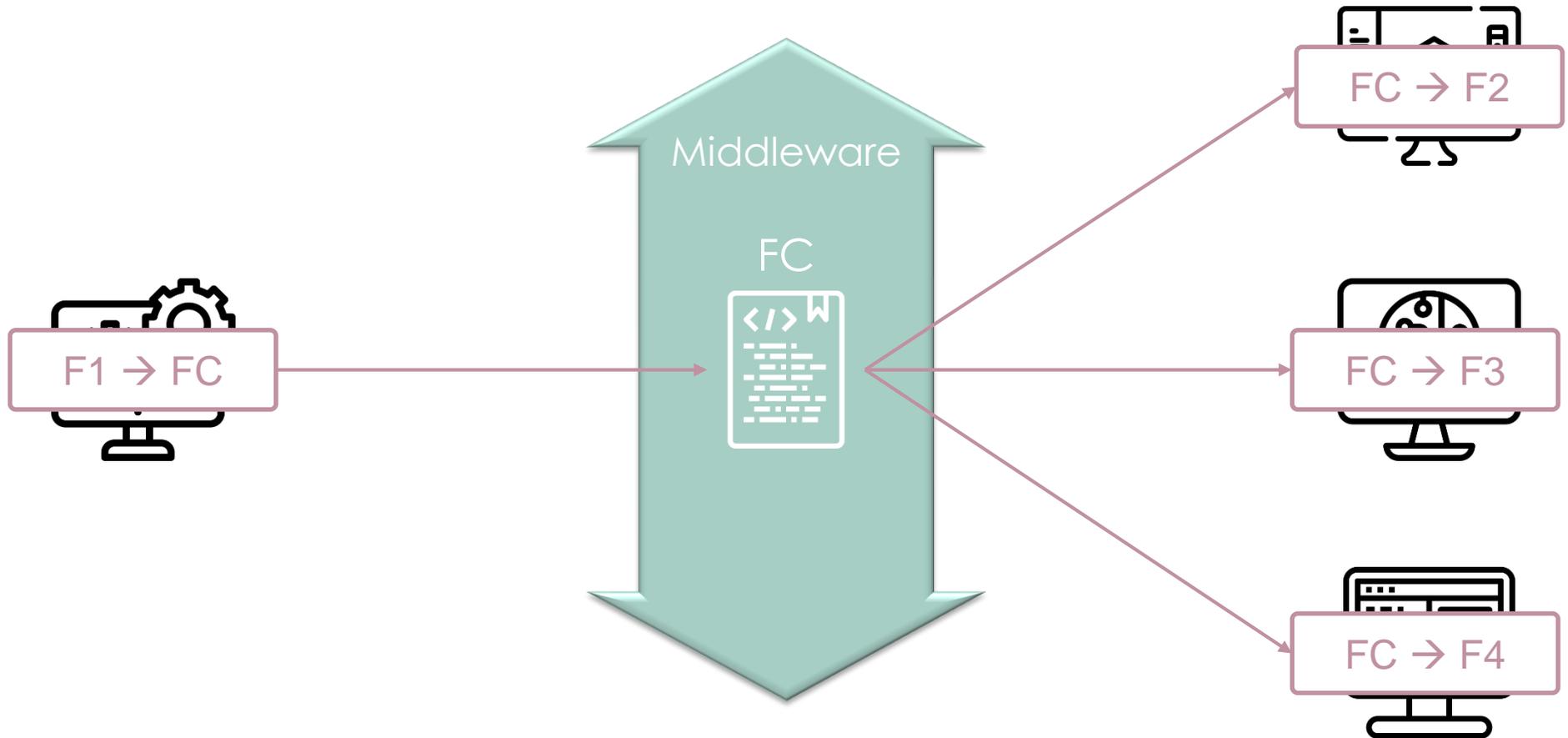


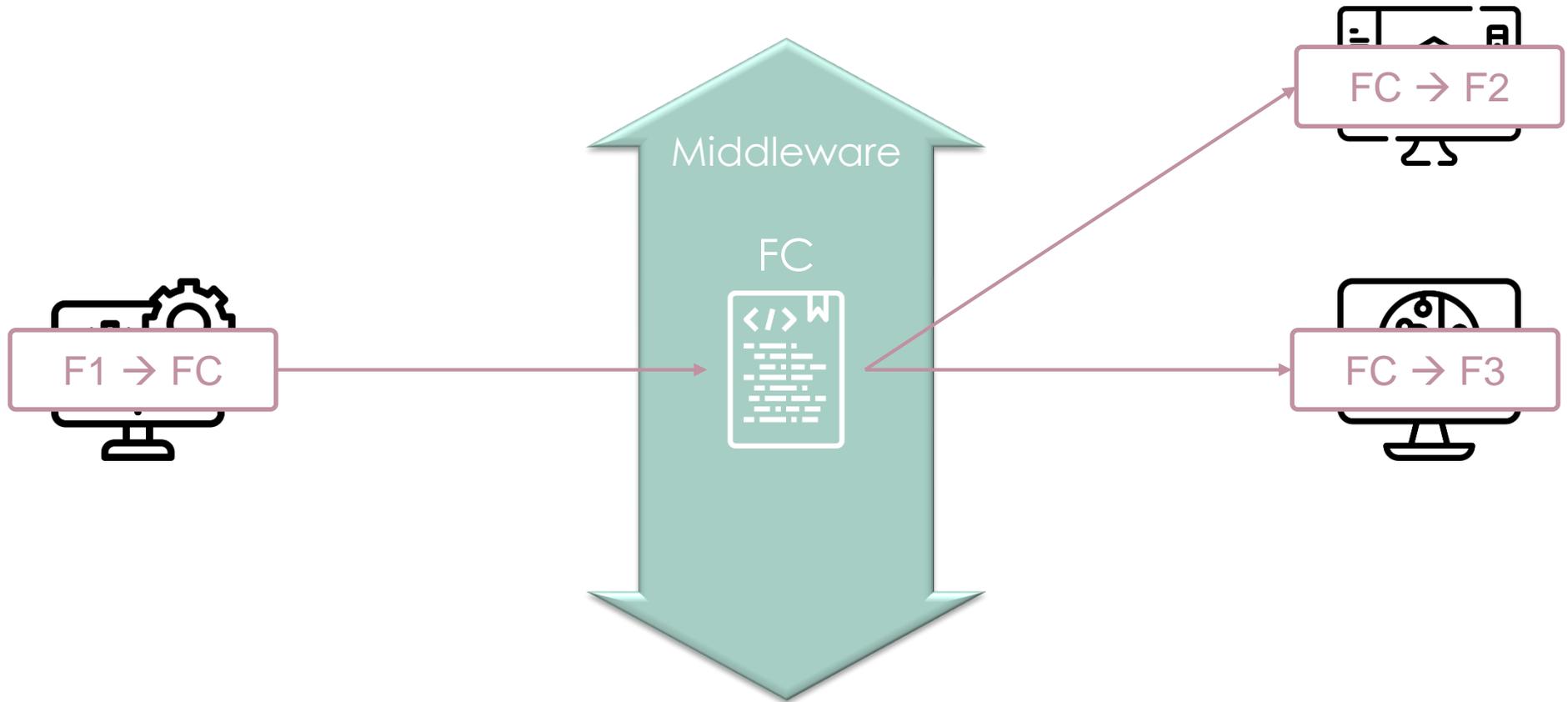
Modèle **physique** de données

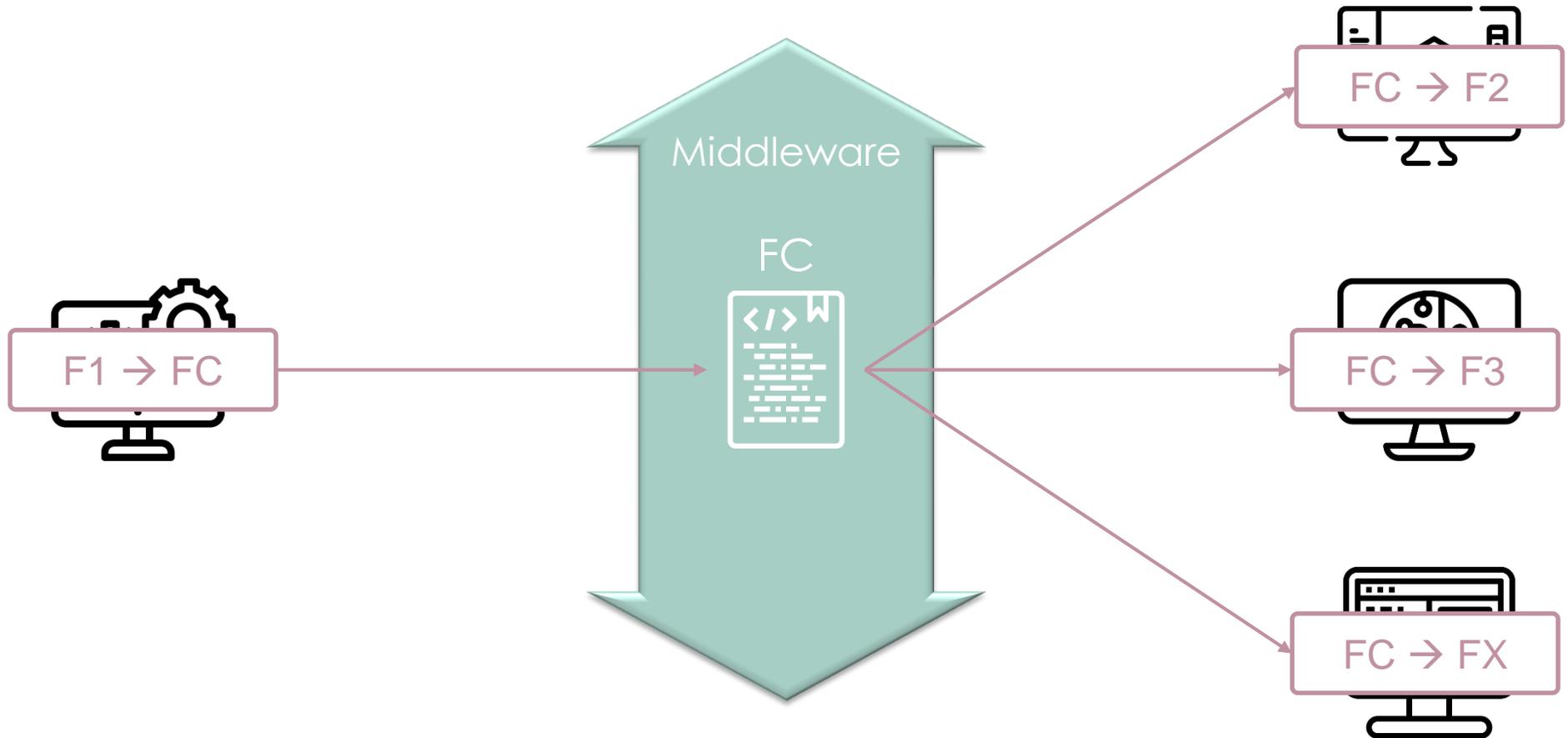
Technologie

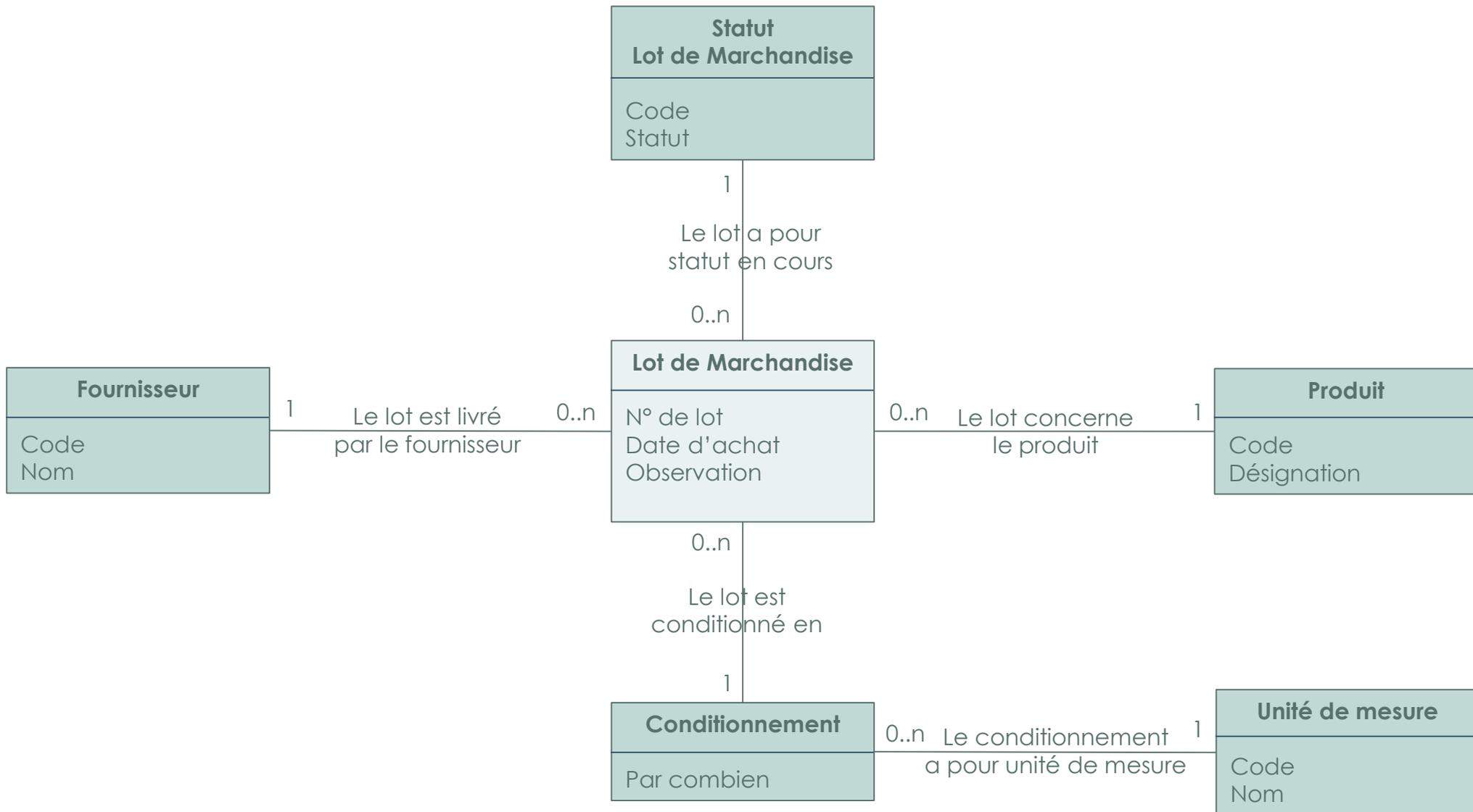




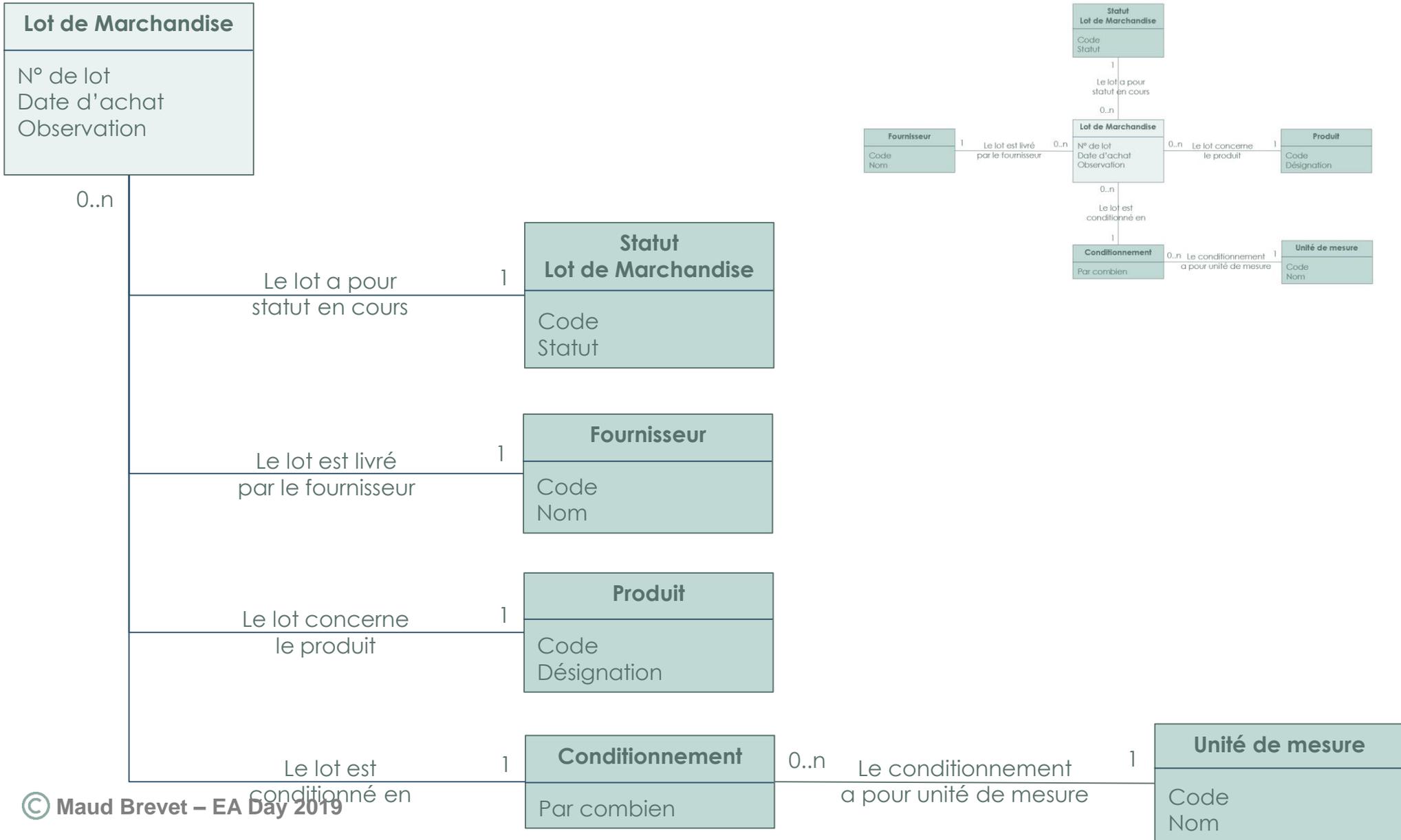


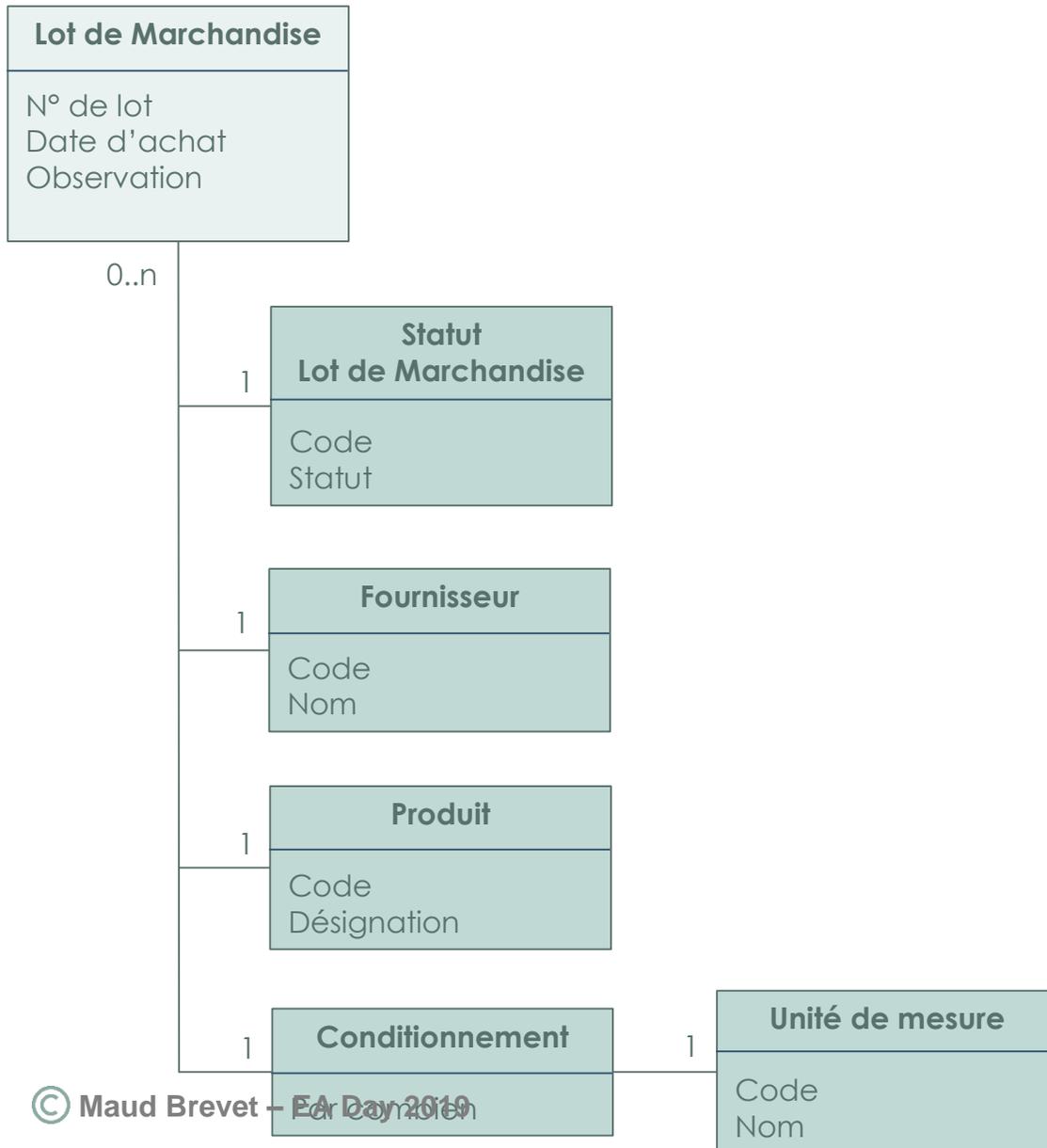






# De conceptuel à canonique





## <Lot de Marchandise>

### 1 <Statut>

Code

Statut

</Statut>

### 1 <Fournisseur>

Code

Nom

</Fournisseur>

### 1 <Produit>

Code

Désignation

</Produit>

### 1 <Conditionnement>

Par combien

#### 1 <Unité de Mesure>

Code

Nom

</Unité de Mesure>

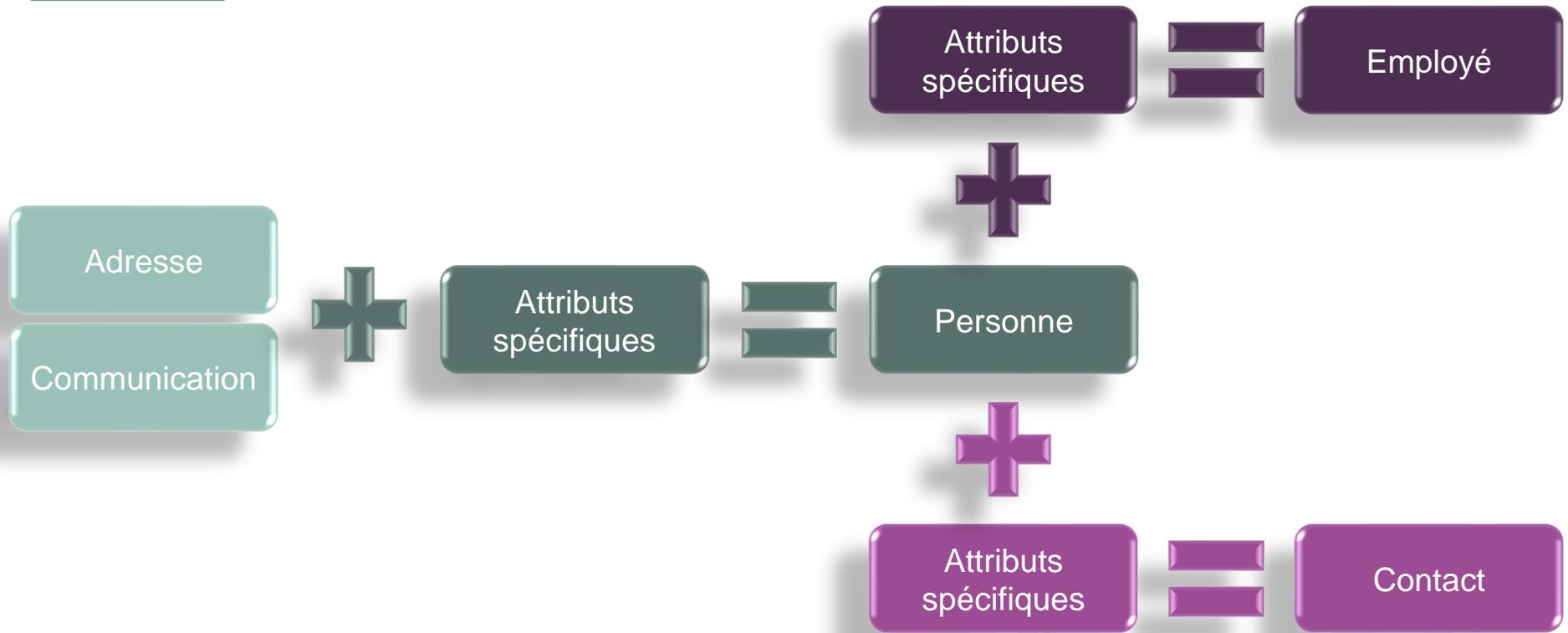
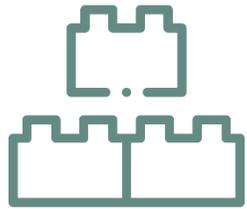
</Conditionnement>

</Lot de Marchandise>

## Exemple (extrait)

```
<xsd:element name = "Address" type = "AddressType"/>
<xsd:complexType name = "AddressType">
  <xsd:sequence>
    <xsd:element ref = "AddressLine" maxOccurs = "unbounded"/>
    <xsd:element ref = "City"/>
    <xsd:element ref = "CountryRegionCode"/>
    <xsd:element ref = "PostalCode" minOccurs = "0"/>
    <xsd:element ref = "CountryCode" minOccurs = "0"/>
    <xsd:element ref = "County" minOccurs = "0"/>
    <xsd:element ref = "CountyFIPSCode" minOccurs = "0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:element name = "AddressLine" type = "AddressLineType"/>
<xsd:simpleType name = "AddressLineType">
  <xsd:restriction base = "xsd:string"/>
</xsd:simpleType>
```

# De simplicité à complexité



Simplicité

Complexité

Définir des **formats pivots**

→ Transformer une vue métier en un **format d'échange standard**

Permettre l'**interopérabilité**

→ Permettre à des **systèmes hétérogènes** de **dialoguer** entre eux

Rester **agnostique** des solutions

→ Construire un **format métier stable** dans le temps

Définir un **GUID**

→ Reconnaître de manière **unique** une donnée dans le SI, **dans l'espace et dans le temps**

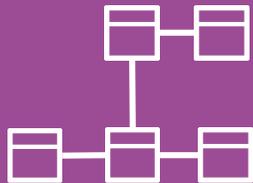
# Représentations de la donnée

## Architecture d'Entreprise



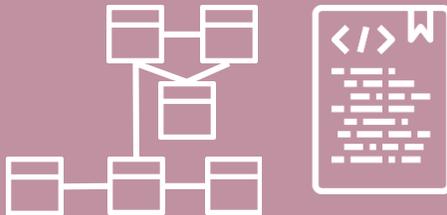
Concept métier

Métier



Modèle **conceptuel** de données

Fonction



Modèle **logique** de données

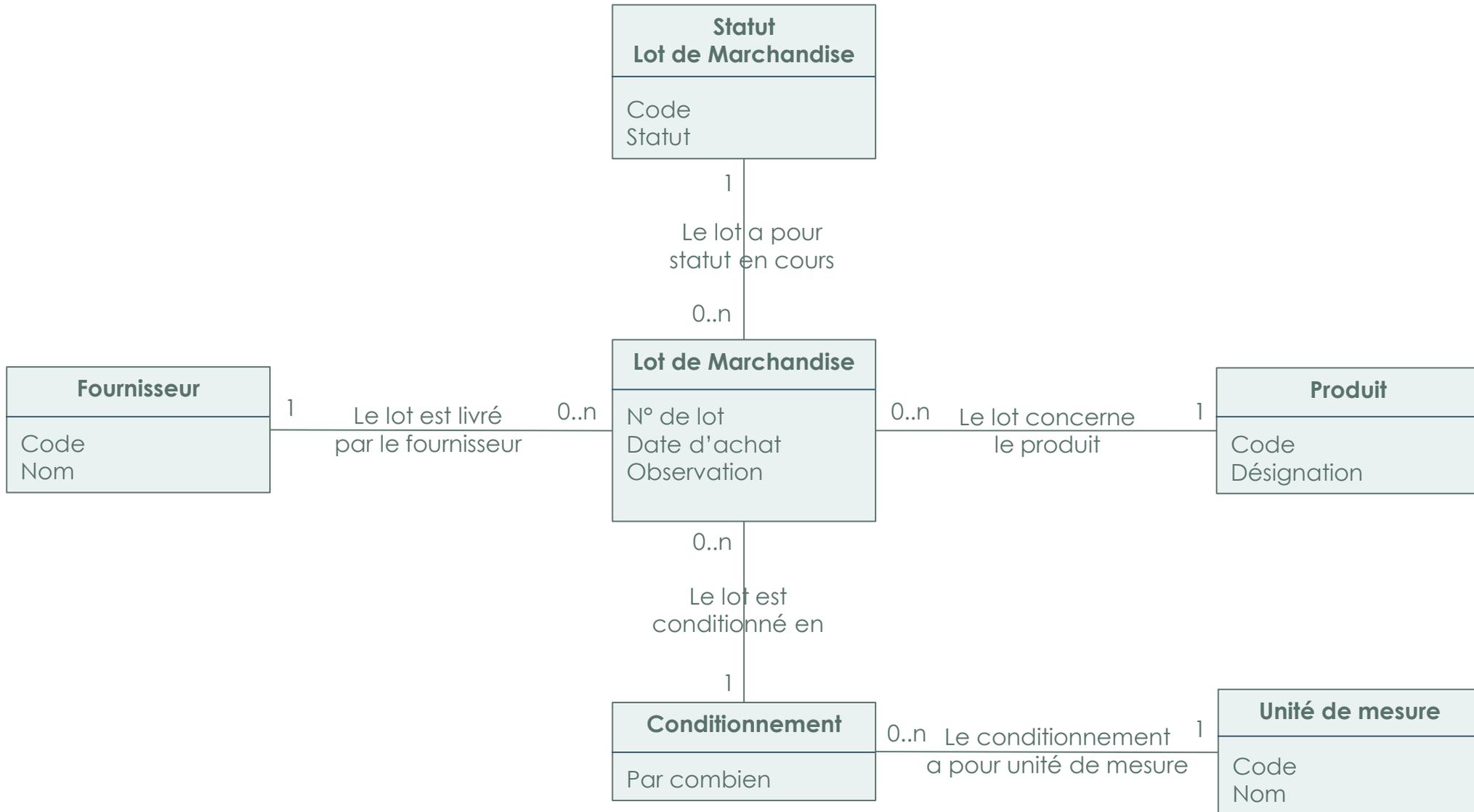
Modèle **canonique** de données

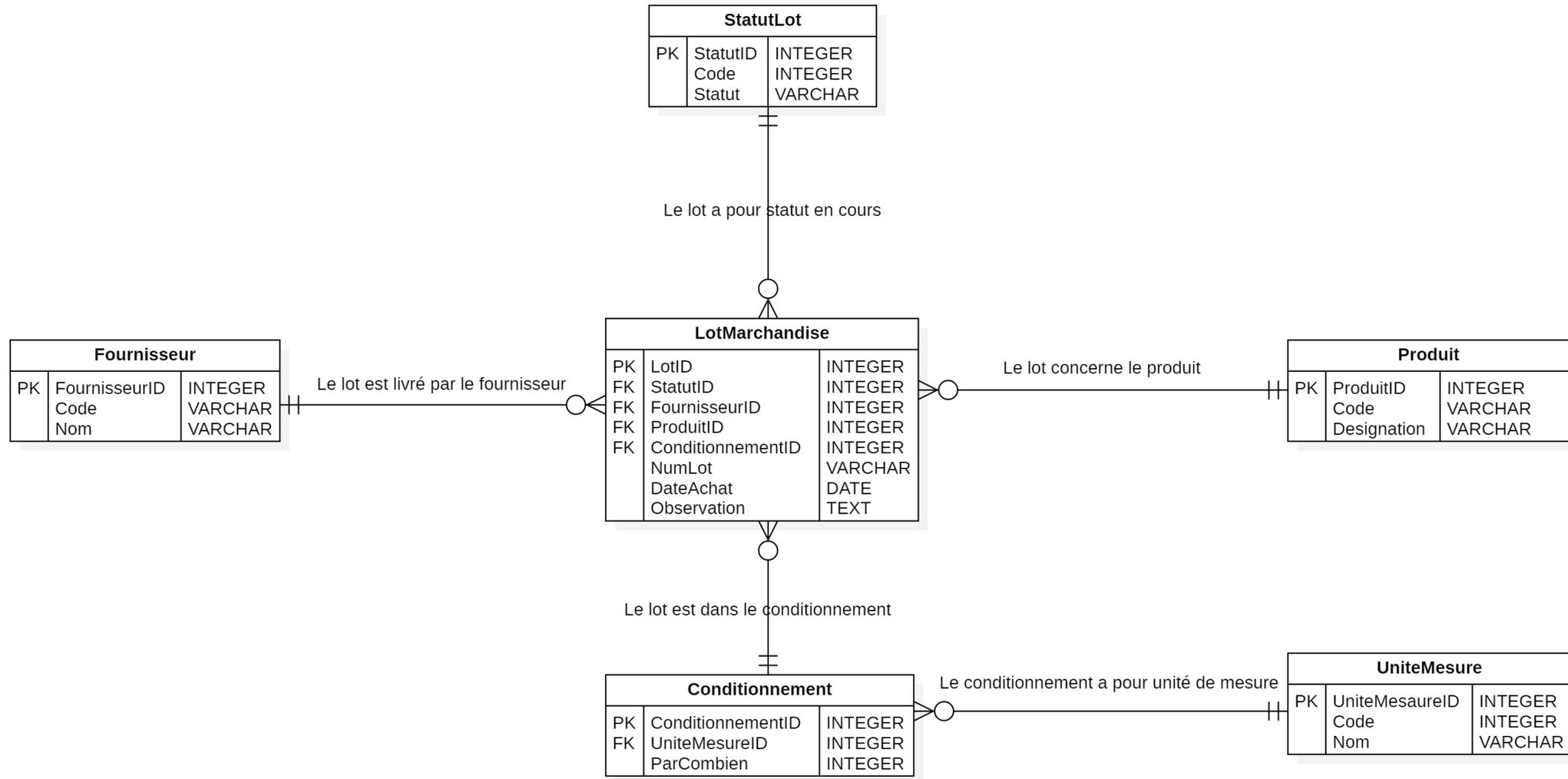
Application

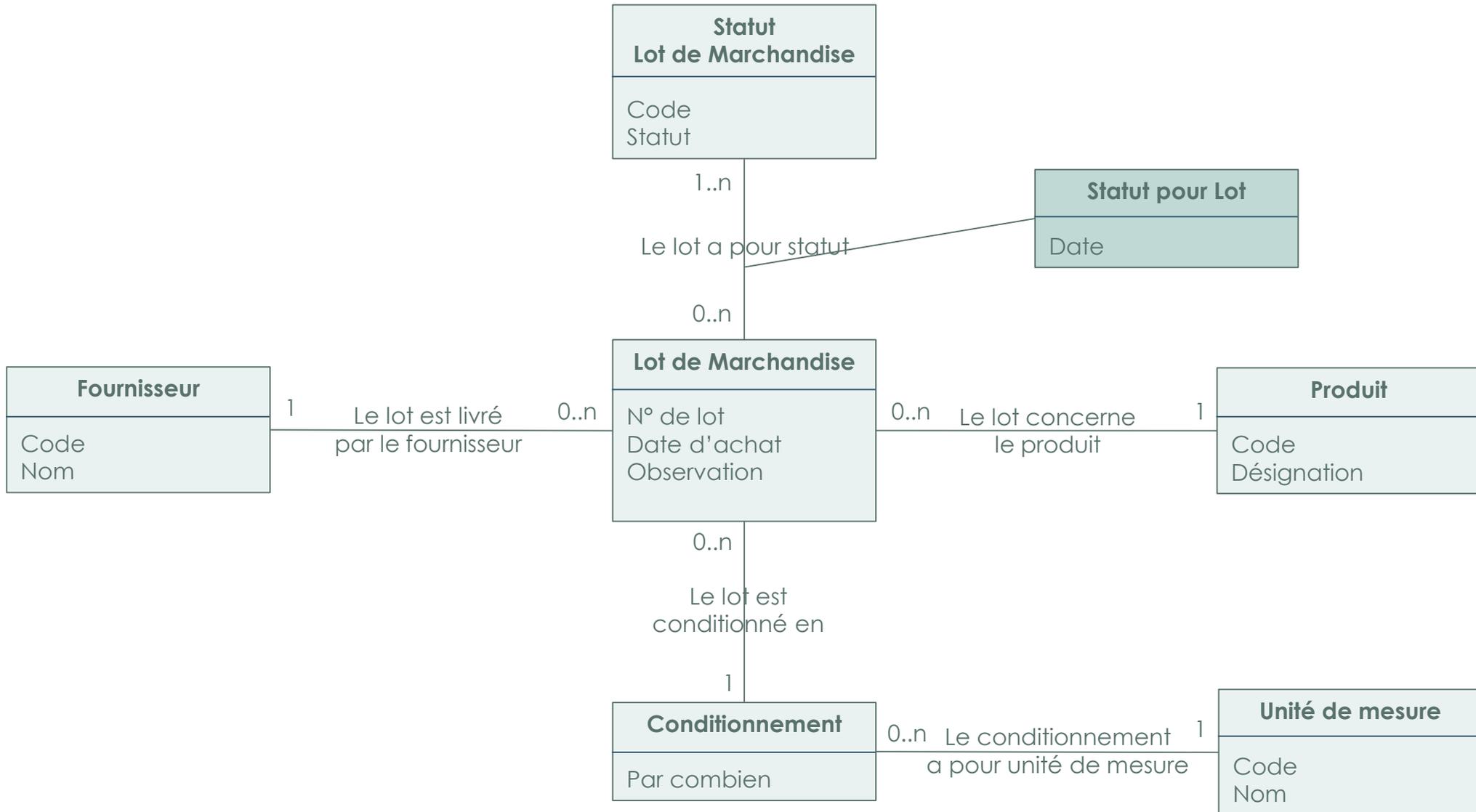


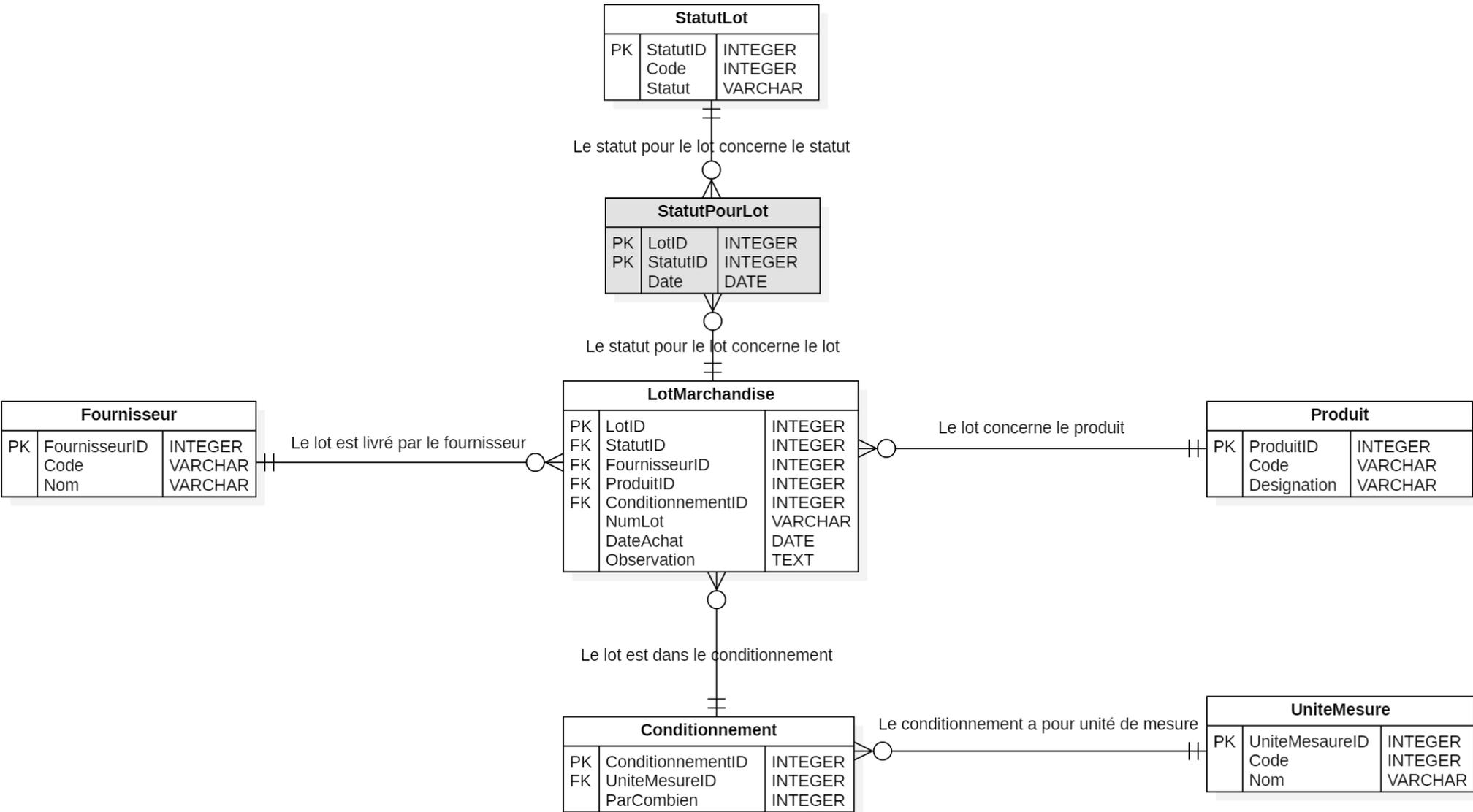
Modèle **physique** de données

Technologie









## Préparer l'implémentation

→ Traduire un modèle conceptuel en **vue d'une implémentation** en modèle physique

## Construire une étape intermédiaire

→ Décrire la **structure** dans laquelle seront **stockées les données** dans une future base de données

## Rester agnostique des solutions

→ Construire un modèle logique **indépendant** de toute solution

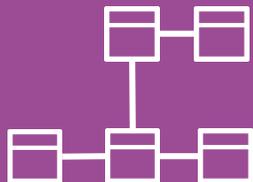
# Représentations de la donnée

## Architecture d'Entreprise



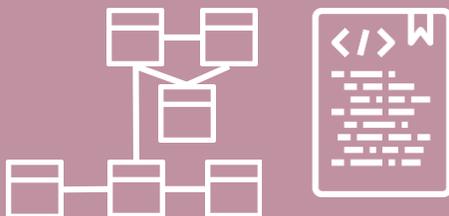
Concept métier

Métier



Modèle **conceptuel** de données

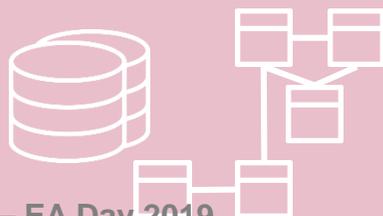
Fonction



Modèle **logique** de données

Modèle **canonique** de données

Application



Modèle **physique** de données

Technologie



Un modèle physique **implémente un modèle logique** pour une **solution spécifique**

**Limite de responsabilité** de l'architecture d'entreprise

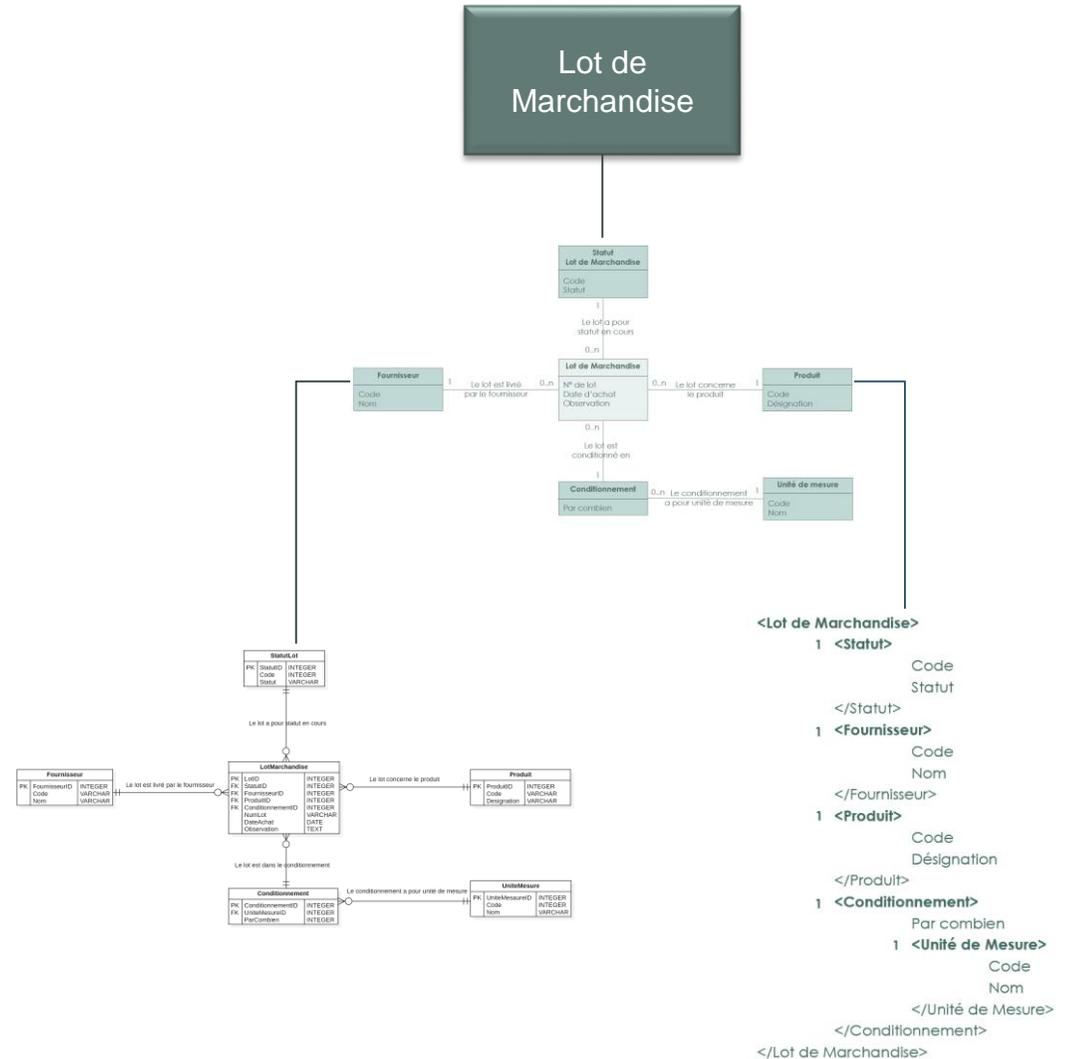
→ **Pas de construire le modèle physique** d'une solution

→ **Vérifier la conformité entre** le modèle **physique** et le modèle **logique**

Chaque représentation a un sens

Ne rater aucune étape

Obtenir un système d'information robuste, flexible, extensible, modulaire & évolutif



# Enterprise Architecture & Agile at a glance

