

# Analyse Projet BDD

## Contents

<b>1</b>	<b>Analyse du problème</b>	<b>2</b>
1.1	Hypothèses . . . . .	2
1.2	Contraintes . . . . .	3
1.3	Diagramme Entités/Associations . . . . .	4
<b>2</b>	<b>Modèle Relationnel</b>	<b>5</b>
2.1	Traduction des entités simples . . . . .	5
2.2	Sous-types d'entités . . . . .	5
2.3	Type d'entité faible . . . . .	5
2.4	Types d'associations . . . . .	5
2.4.1	Cardinalité 1..1 . . . . .	5
2.4.2	Cardinalité 0..1 . . . . .	5
2.4.3	Cardinalité ?..* . . . . .	5
<b>3</b>	<b>Analyse des fonctionnalités</b>	<b>5</b>
<b>4</b>	<b>Bilan du projet</b>	<b>5</b>
<b>5</b>	<b>Mode d'emploi</b>	<b>5</b>

# 1 Analyse du problème

## 1.1 Hypothèses

Pour la conception, nous avons fait les choix suivants:

1. Le statut des commandes n'est pas dans l'entité mère COMMANDES mais dans ses sous-entités afin d'éviter d'avoir des statuts non cohérents pour un type de commande donné.
2. Nous avons rajouté le statut "terminée" qui indique qu'une commande a effectivement été effectuée et que l'on peut donner une évaluation
- 3.
- 4.

Nous sommes arrivés avec les contraintes suivantes :

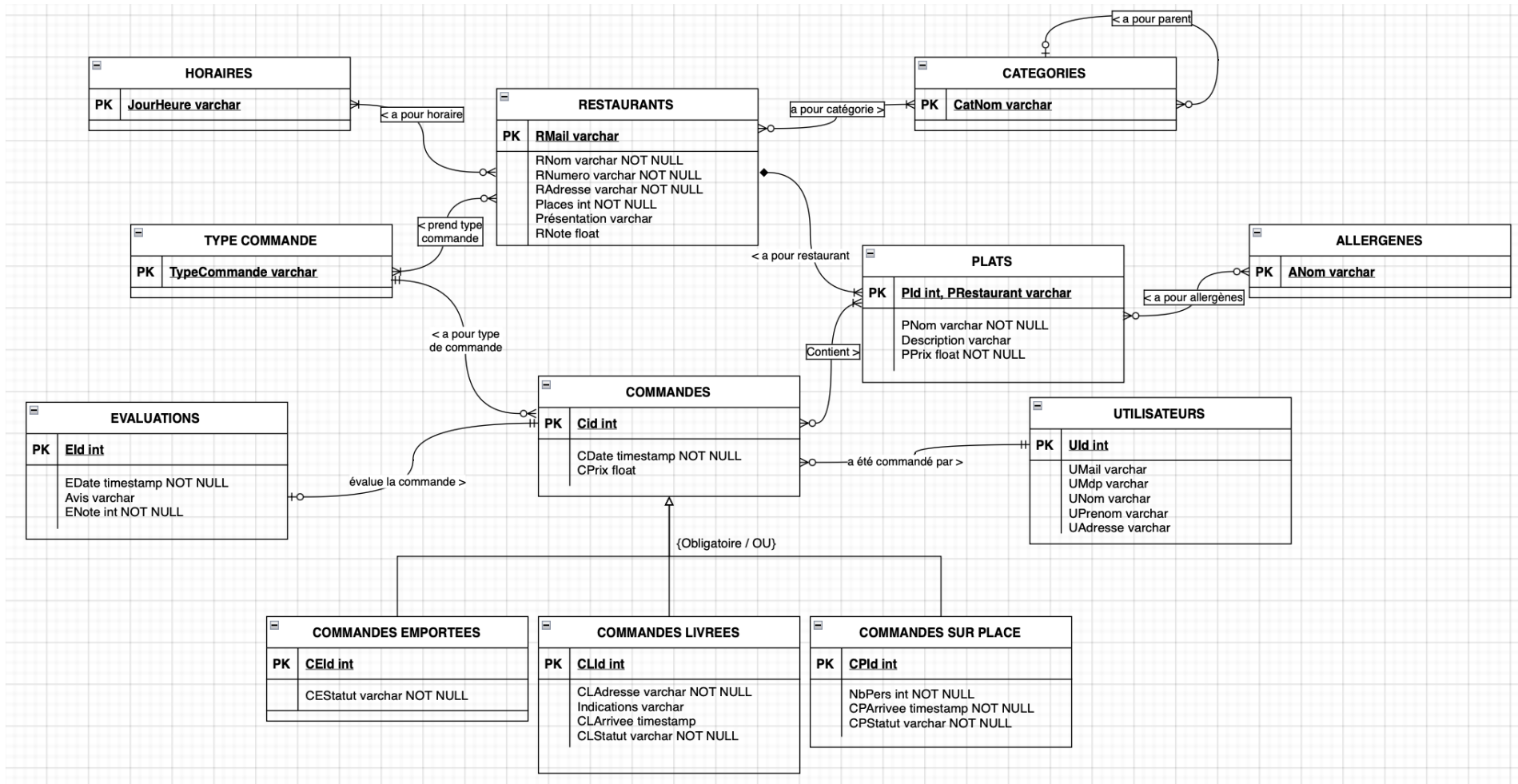
## 1.2 Contraintes

3

DF	C. Valeurs	C. Contextuelles	C. Multiplicité
RMail $\rightarrow$ RNom, RNumero, RAdresse, Places, Presentation, RNote	RType $\in \{\text{livraison, emporter, place}\}$  Places $> 0$  RNote $\in [0, 5]$	$\sum \text{nbPers} \leq \text{Places}$ pour un restaurant et ses commandes associées Ext(CEId) $\cap$ Ext(CPId) $\cap$ Ext(CLId) $= \emptyset$ Ext(CEId) $\cup$ Ext(CPId) $\cup$ Ext(CLId) $= \text{Ext}(\text{CId})$ CPArrivee $\in \text{JourPlage}$ pour un restaurant et ses commandes associées CDate donne JourPlage pour toute commande CDate $\leq$ EDate	RMail $\twoheadrightarrow$ JourPlage RMail $\twoheadrightarrow$ TypeCommande  RMail $\twoheadrightarrow$ PId  RMail $\twoheadrightarrow$ CatNom  (PId, RMail) $\nleftrightarrow$ ANom CId $\twoheadrightarrow$ (PId, RMail)
(PId, Restaurant) $\rightarrow$ PNom, PDescription, PPrix	PPrix $> 0$	EId $\Rightarrow$ CId.statut = {terminee}	CId $\rightarrow$ TypeCommande
U_Id $\rightarrow$ UMail, Umdp, UNom, UPrenom, UAdresse			
CId $\rightarrow$ CDate, CPrix	CPrix $> 0$		
CLId $\rightarrow$ CLAdresse, Indications, CLArrivee, CLStatut	CLStatut $\in \{\text{attente, validée, en livraison, annuleeC, annuleeR, terminee}\}$		
CEId $\rightarrow$ CESTatut	CEStatut $\in \{\text{attente, validée, disponible, annuleeC, annuleeR, terminee}\}$		
CPId $\rightarrow$ NbPers, CPArrivee, CPStatut	CPStatut $\in \{\text{attente, validée, annuleeC, annuleeR, terminee}\}$ NbPers $> 0$	CType $\in \text{TypeCommande}$ pour un CId donné et le RMail associé	CId $\rightarrow$ U_Id  CId $\leftrightarrow$ EId CatNom $\leftrightarrow$ CatNom
EId $\rightarrow$ EDate, Avis, ENote	ENote $\in [0..5]$		
	JourHeure $\in \{\text{LM, LS, MM, MS, MeM, MeS, JM, JS, VM, VS, SM, SS, DM, DS}\}$		

### 1.3 Diagramme Entités/Associations

On obtient ensuite le diagramme Entité-Relation suivant:



## 2 Modèle Relationnel

Nous avons appliqué l'algorithme vu en cours pour obtenir le schéma relationnel.

### 2.1 Traduction des entités simples

Nous avons identifié les entités simples suivantes:

- Restaurant(RMail, RNom, RNum, RAdresse, Places, Présentation, RNote)
- Commandes(Cid, CDate, CPrix, **Uid**, **TypeCommande**)
- Utilisateurs(U\_id, UMail, UMdp, UNom, UPrenom, UAdresse)
- Evaluation(Eid, EDate, Avis, ENote, **Cid**)
- Horaires(JourPlage)
- TypesCommande(TypeCommande)
- Categories(CatNom)
- Allergenes(ANom)

### 2.2 Sous-types d'entités

Pour une question d'espace occupé nous avons décidé de traduire l'héritage des différentes catégories des commandes par référence. Contrairement à l'unification cela nous permet de ne pas avoir à gérer les contraintes avec l'application.

- CommandesEmportees(CEid, CESTatut)
- CommandesLivrees(CLid, CLAdresse, Indications, CLArrivee, CLStatut)
- CommandesSurPlace(CPid, NbPers, CPArrivee, CPStatut)

### 2.3 Type d'entité faible

- Plats(Pid, PRestaurant, PNom, Description, PPrix)

**Contrainte induite par la traduction:** Vérifier que tout restaurant a au moins un plat

### 2.4 Types d'associations

#### 2.4.1 Cardinalité 1..1

#### 2.4.2 Cardinalité 0..1

#### 2.4.3 Cardinalité ?..\*

## 3 Analyse des fonctionnalités

## 4 Bilan du projet

## 5 Mode d'emploi