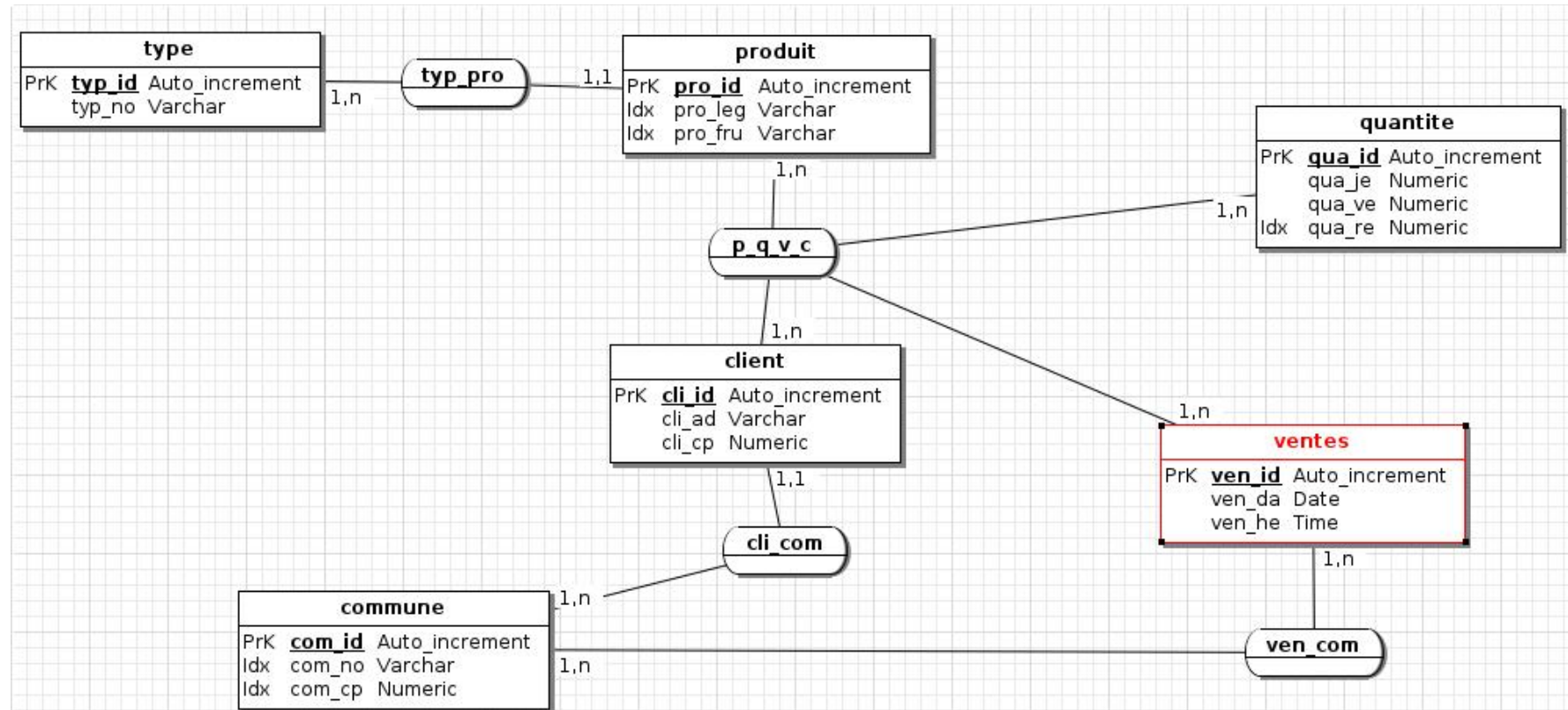


Compétence 2 : Concevoir une base de données FAIS PAS LE POIREAU

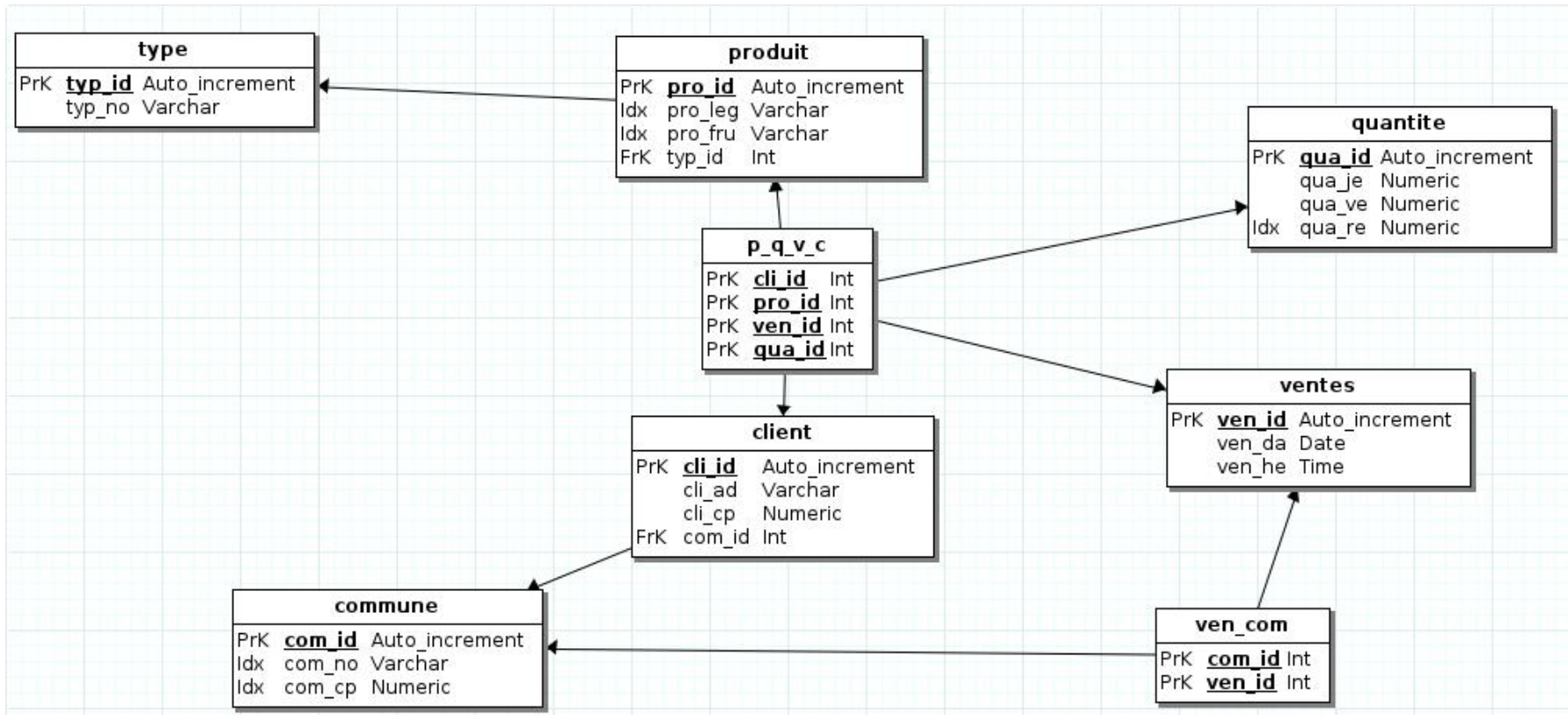
1. DICTIONNAIRE DE DONNÉES

Description de la donnée	Type	Longueur	Exemples	Nom de la colonne	Contraintes
identifiant de produit	numérique	25	1	pro_id	clé primaire, Auto-increment, Not null
nom des fruits	alphanumérique	50	cerise	pro_fru	not null, Index
nom des légumes	alphanumérique	50	tomate	pro_leg	not null Index
identifiant de la quantité	numérique	25	2	qua_id	clé primaire, Auto-increment, Not null
quantité jetée	numérique	25	14	qua_je	not null
quantité vendue	numérique	25	12	qua_ve	not null
quantité restante	numérique	25	45	qua_re	not null, Index, Nb=5 = alerte
identifiant du type de produit	numérique	25	3	typ_id	Auto-increment, Not null
nom du type de produit	alphanumérique	25	fruit	typ_no	not null
identifiant de la vente	numérique	25	2	ven_id	clé primaire, Auto-increment, Not null
date de la vente	date	10	23/03/18	ven_da	not null
heure de la vente	time	10	12:12:00	ven_he	not null
identifiant du client	numérique	25	1	cli_id	clé primaire, Auto-increment, Not null
adresse du client	alphanumérique	100	4 avenue des pyrénées	cli_ad	
code postal du client	numérique	5	O9100	cli_cp	
identifiant de la commune	numérique	25	1	com_id	clé primaire, Auto-increment, Not null
nom de la commune	alphanumérique	50	pamiers	com_no	not null, index
code postal de la commune	numérique	5	O9100	com_cp	not null, index

2. MCD



3. MLD



4. SGBD CHOISI



5. MPD et JUSTIFICATION COLONNES INDEXÉES

Colonnes com_no et com_cp : visualiser rapidement les 5 communes les plus consommatrices et faire le bilan pour chaque commune.

Colonnes pro_fru et pro_leg : afin de pouvoir contacter au plus vite son fournisseur lorsque l'alerte se déclenche et savoir les noms des fruits et légumes manquants.

Colonnes qua_re : contacter son fournisseur rapidement lorsque le stock passe en dessous des 5.

-- Script Postgre

-- Table: produit

```
-----  
CREATE TABLE public.produit(  
    pro_id SERIAL NOT NULL ,  
    pro_leg VARCHAR (50) NOT NULL ,  
    pro_fru VARCHAR (50) NOT NULL ,  
    typ_id INT NOT NULL ,  
    CONSTRAINT prk_constraint_produit PRIMARY KEY (pro_id)  
)WITHOUT OIDS;
```

-- Table: quantite

```
-----  
CREATE TABLE public.quantite(  
    qua_id SERIAL NOT NULL ,  
    qua_je NUMERIC (50,0) NOT NULL ,  
    qua_ve NUMERIC (50,0) NOT NULL ,  
    qua_re NUMERIC (25,0) NOT NULL ,  
    CONSTRAINT prk_constraint_quantite PRIMARY KEY (qua_id)  
)WITHOUT OIDS;
```

-- Table: type

```
-----  
CREATE TABLE public.type(  
    typ_id SERIAL NOT NULL ,  
    typ_no VARCHAR (25) NOT NULL ,  
    CONSTRAINT prk_constraint_type PRIMARY KEY (typ_id)  
)WITHOUT OIDS;
```

-- Table: ventes

```
-----  
CREATE TABLE public.ventes(  
    ven_id SERIAL NOT NULL ,  
    ven_da DATE NOT NULL ,  
    ven_he TIMETZ NOT NULL ,  
    CONSTRAINT prk_constraint_ventes PRIMARY KEY (ven_id)  
)WITHOUT OIDS;
```

-- Table: client

```
-----  
CREATE TABLE public.client(  
    cli_id SERIAL NOT NULL ,  
    cli_ad VARCHAR (100) ,  
    cli_cp NUMERIC (5,0) ,  
    com_id INT NOT NULL ,  
    CONSTRAINT prk_constraint_client PRIMARY KEY (cli_id)  
)WITHOUT OIDS;
```

-- Table: commune

```
-----  
CREATE TABLE public.commune(  
    com_id SERIAL NOT NULL ,  
    com_no VARCHAR (50) NOT NULL ,  
    com_cp NUMERIC (5,0) NOT NULL ,  
    CONSTRAINT prk_constraint_commune PRIMARY KEY (com_id)  
)WITHOUT OIDS;
```

-- Table: p_q_v_c

```
-----  
CREATE TABLE public.p_q_v_c(  
    cli_id INT NOT NULL ,  
    pro_id INT NOT NULL ,  
    ven_id INT NOT NULL ,  
    qua_id INT NOT NULL ,  
    CONSTRAINT prk_constraint_p_q_v_c PRIMARY KEY (cli_id,pro_id,ven_id,qua_id)  
)WITHOUT OIDS;
```

-- Table: ven_com

```
-----  
CREATE TABLE public.ven_com(  
    com_id INT NOT NULL ,  
    ven_id INT NOT NULL ,  
    CONSTRAINT prk_constraint_ven_com PRIMARY KEY (com_id,ven_id)  
)WITHOUT OIDS;
```

```
CREATE INDEX produit_pro_leg_idx ON produit (pro_leg);  
CREATE INDEX produit_pro_fru_idx ON produit (pro_fru);  
CREATE INDEX quantite_qua_re_idx ON quantite (qua_re);  
CREATE INDEX commune_com_no_idx ON commune (com_no);  
CREATE INDEX commune_com_cp_idx ON commune (com_cp);
```

```
ALTER TABLE public.produit ADD CONSTRAINT FK_produit_typ_id FOREIGN KEY (typ_id) REFERENCES public.type(typ_id);  
ALTER TABLE public.client ADD CONSTRAINT FK_client_com_id FOREIGN KEY (com_id) REFERENCES public.commune(com_id);  
ALTER TABLE public.p_q_v_c ADD CONSTRAINT FK_p_q_v_c_cli_id FOREIGN KEY (cli_id) REFERENCES public.client(cli_id);  
ALTER TABLE public.p_q_v_c ADD CONSTRAINT FK_p_q_v_c_pro_id FOREIGN KEY (pro_id) REFERENCES public.produit(pro_id);  
ALTER TABLE public.p_q_v_c ADD CONSTRAINT FK_p_q_v_c_ven_id FOREIGN KEY (ven_id) REFERENCES public.ventes(ven_id);  
ALTER TABLE public.p_q_v_c ADD CONSTRAINT FK_p_q_v_c_qua_id FOREIGN KEY (qua_id) REFERENCES public.quantite(qua_id);  
ALTER TABLE public.ven_com ADD CONSTRAINT FK_ven_com_com_id FOREIGN KEY (com_id) REFERENCES public.commune(com_id);
```

```
ALTER TABLE public.ven_com ADD CONSTRAINT FK_ven_com_ven_id FOREIGN KEY (ven_id) REFERENCES public.ventes(ven_id);
```