

Prolog

Opérateurs :

=	Unification des termes à droite et à gauche
==	Identité syntaxique
:=	Egalité arithmétique
is	Instanciation arithmétique
\== ou \=	Différence syntaxique
=\<	Différence arithmétique
+	Addition
-	Soustraction
*	Produit
/	Division réelle
//	Division entière
mod	Reste de la division euclidienne
< > <= >=	Comparaison arithmétique

Prolog

Opérateurs

- Unification :
?- X+1=2+1.
X = 2.

?- 2+X=3+1.
false.
- Identité syntaxique :
?- X==X.
true.

?- dad==mam.
false.

?- "dad"=="dad".
true.

?- dad==dad.
true.
- Egalité arithmétique :
?- 1+3:=4.
true.

?- 1+3:=3+1.
true.
- Instanciation arithmétique :
?- X is 3+4.
X = 7.

?- X is +(3,4).
X = 7.

?- X is 3*4+4*5.
X = 32.

?- X is +(*(3,4),*(4,5)).
X = 32.
- Différence syntaxique :
?- X\=Y. (variables !)
false.

?- dad\<=mam.
true.

?- dad\<==mam.
true.

?- dad\<=dad.
false.
- Différence arithmétique :
?- 3=\=6.
true.

?- 3=\=3.
false.

Prolog

Structure d'un programme :

- Faits
- Règles
- Buts (requêtes)

Symboles utilisés :

Prédicat (ou fonction, constante)	Mot ou lettre en minuscule
Variable	Mot en majuscule (ou commençant par un majuscule ou par _)
:-	Si (if) Exp. <code>individu(X):-personne(X)</code> . Se lit : X est un individu si X est une personne
,	Et (and)
;	Ou (or)
not	Négation (not)
_	Variable anonyme (\forall)

Prolog

- Tous les chiens sont des animaux $\equiv \forall x \text{ Chien}(x) \Rightarrow \text{Animal}(x)$
- Tous les animaux vont mourir $\equiv \forall x \text{ Animal}(x) \Rightarrow \text{Mourir}(x)$
- Fido est un chien $\equiv \text{Chien}(\text{Fido})$

Déductions :

Fido est un animal $\equiv \text{Animal}(\text{Fido})$

Fido va mourir $\equiv \text{Mourir}(\text{Fido})$

Prolog

Individu / Personne :

- **Créer des personnes (définir un âge pour chacun).**
- **Un individu est une personne quelque soit son âge.**
- **Un mineur est toute personne ayant moins de 18 ans.**

Code Prolog (voir le fichier : `personne.pl`)

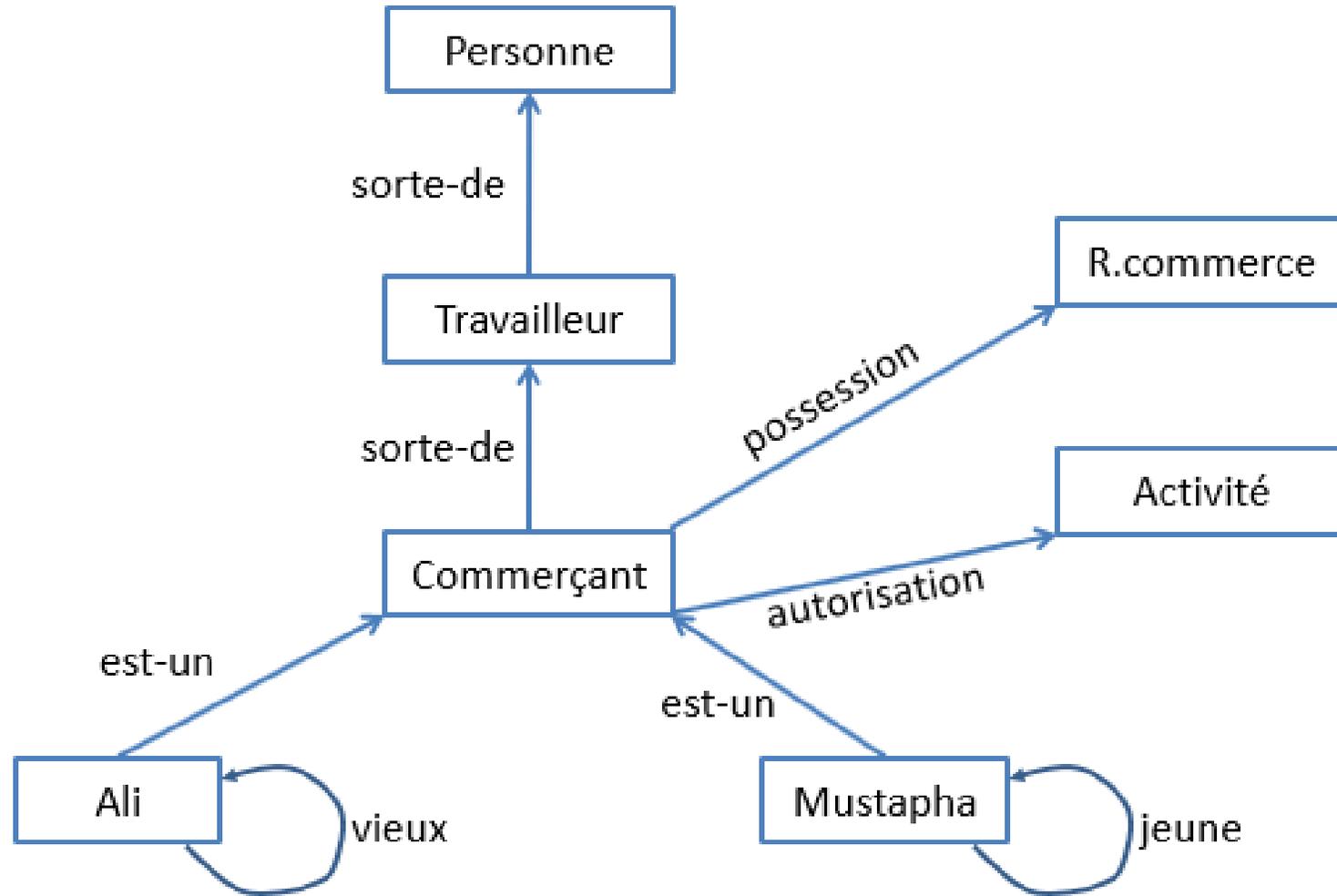
Prolog

Arbre familial (family tree) :

- **Utiliser les concepts :**
 - **father, mother, parent, grandparent...**

Code Prolog (voir le fichier : family.pl)

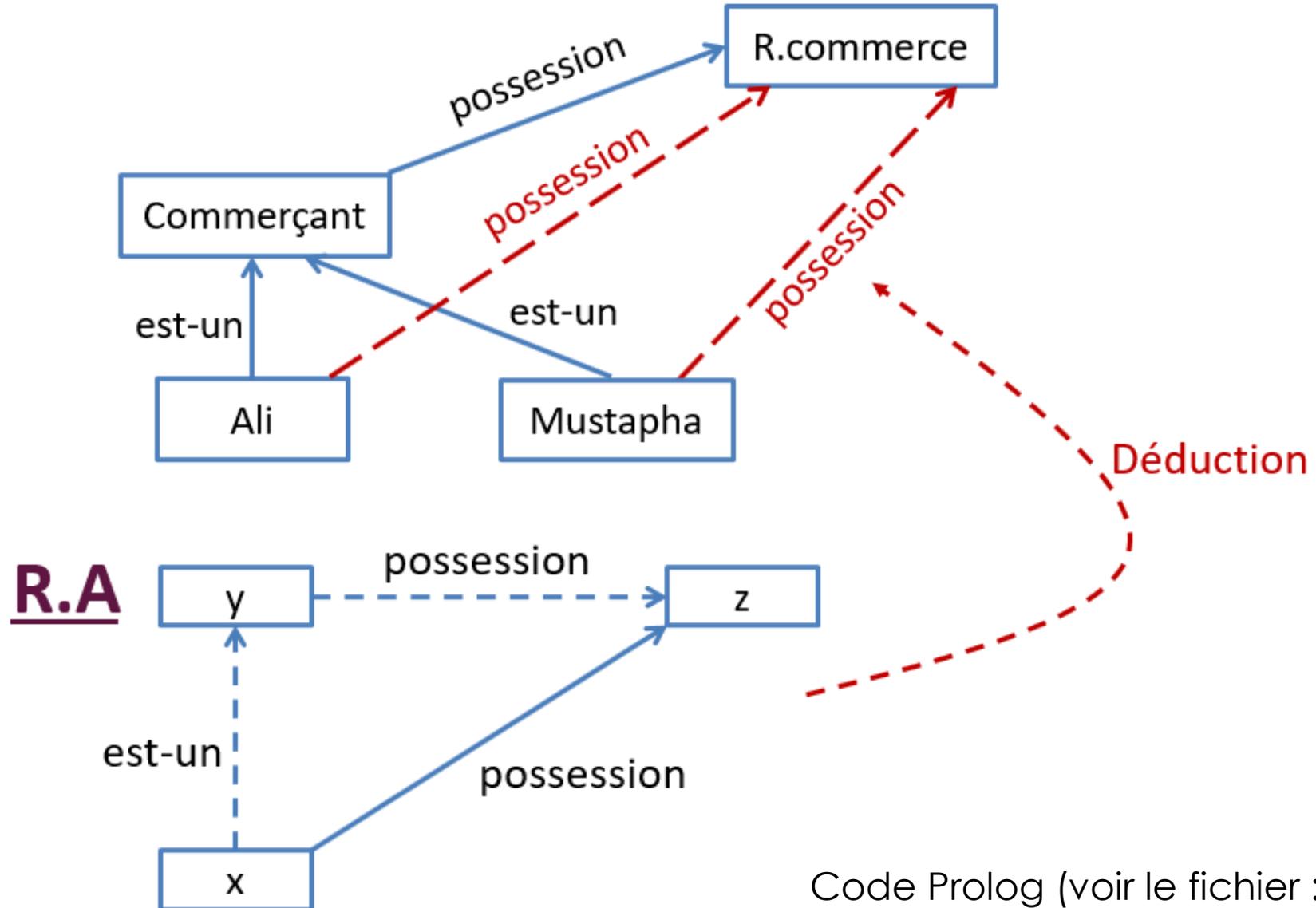
Prolog



Code Prolog (voir le fichier : commerce.pl)

Prolog

Qui a un registre de commerce ?



Code Prolog (voir le fichier : commerce.pl)