

# **Système Expert à base de règle : Conseiller canin**

## **Audrey Szczepanski - Iker Tardio**

### **Introduction**

Ce TP nous a permis de réaliser un système expert d'ordre 0+. Un système expert est un programme informatique qui « raisonne sur des problèmes dont la résolution repose sur une expertise humaine dans un domaine délimité. » (Edward Feigenbaum). L'utilisation d'un tel système est justifié quand la résolution du problème ne connaît pas d'algorithme connu ou de modèle mathématique. Pour raisonner, un SE se compose de trois parties : une base de règles qui indique le raisonnement à suivre, une base de faits que donne l'utilisateur, et un moteur d'inférence qui fait fonctionner les règles à partir de ces faits pour obtenir de nouveaux faits jusqu'à atteindre la réponse à la question posée. Le SE proposé est d'ordre 0+ : les faits sont représentés comme des couples (opérateur objet valeur).

Nous présenterons tout d'abord notre problématique et l'expertise, puis la représentation des connaissances, faits et règles, ainsi que des jeux d'essais et enfin la représentation lisp utilisée pour programmer notre système expert.

Ce rapport sera associé à un fichier contenant le code lisp de notre système expert avec plusieurs scénarios d'utilisation.

### **Question 1 - Problématique**

Nous avons choisi d'implémenter un système expert dans le domaine canin.

La problématique associée à ce système expert serait "Quels races de chien conviendraient à un individu donné ?". Le but serait de définir en fonction des caractéristiques de vie d'un individu et des caractéristiques des différentes races de chien, les races de chien dont il serait apte à s'occuper correctement. Cela permettrait à un individu de savoir suivant son mode de vie quels races de chien lui correspondrait le mieux.

La base de fait serait donc constituée des caractéristiques de vie de l'individu comme le nombre d'enfant, la fréquence de pratique d'une activité sportive ou la possession d'un jardin, et des caractéristiques des races de chien, comme le besoin de faire de l'exercice ou la capacité à supporter la solitude. Le but serait une race de chien spécifique, comme un berger allemand ou un bichon frisé.

## Question 2 - Expertise

Pour obtenir des règles, il est nécessaire de procéder à une expertise au préalable afin de pouvoir construire de véritables règles. Une des parties la plus conséquente de ce TP est l'expertise.

Pour construire notre base de connaissance, nous avons utilisé le site <https://wamiz.com/>, qui présente une description des races de chien. Nous nous sommes servi plus particulièrement des parties donnant leurs caractères, leurs aptitudes et leurs conditions de vie.

Cela nous a permis de définir un certain nombre de caractéristiques en lien avec le mode de vie de l'individu.

Nous avons également défini les caractéristiques de l'individu selon ce qui pourrait avoir un impact sur le choix de la race du chien.

### Règles définies pour la base de règles:

R1: SI (caracteristiqueChien = compagnie) ET (exercice? >= moyen) ET (supporteSolitude = non) ALORS (raceDeChien = Boberman)

R2: SI (caracterIndividu = autoritaire) ET (exercice? >= moyen) ET (typeDeChien = extérieur)  
ET (supporteSolitude <= moyen) ALORS (raceDeChien = Beauceron)

R3: SI (caracteristiqueChien = compagnie) ET (supporteSolitude = non) ALORS  
(raceDeChien = BichonFrise)

R4: SI (caracteristiqueChien = compagnie) ET (supporteSolitude <= moyen) ALORS  
(raceDeChien = Carlin)

R5: SI (caracteristiqueChien = compagnie) ET (supporteSolitude = oui) ALORS  
(raceDeChien = Pékinois)

R6: SI (caracteristiqueChien = compagnie) ET (typeDeChien = extérieur) ET (exercice? = oui) ET (supporteSolitude = oui) ALORS (raceDeChien = MalamuteDeLAlaska)

R7: SI (caracterIndividu = autoritaire) ET (caracteristiqueChien = compagnie) ET (exercice? >= moyen) ET (supporteSolitude <= moyen) ALORS (raceDeChien = Chihuahua)

R8: SI (caracterIndividu = autoritaire) ET (caracteristiqueChien = compagnie) ET (exercice? >= moyen) ET (supporteSolitude <= moyen) ALORS (raceDeChien = Leonberger)

R9: SI (caracteristiqueChien = guide) ET (exercice? >= moyen) ET (supporteSolitude = non) ALORS (raceDeChien = Labrador)

R10: SI (caracteristiqueChien = guide) ET (caracterIndividu = autoritaire) ET (exercice? >= oui) ET (supporteSolitude <= moyen) ALORS (raceDeChien = BergerAllemand)

R11: SI (frequenceSport > 1fois/semaine) ALORS (sportif? = oui)

R12: SI (frequenceSport = 1fois/semaine) ALORS (sportif? = moyen)

R13: SI (frequenceSport < 1fois/semaine) ALORS (sportif? = non)

R14: SI (sportif? = oui) ALORS (exercice? = oui)

R15: SI (sportif? = moyen) ALORS (exercice? = moyen)

R16: SI (sportif? = non) ALORS (exercice? = non)

R17: SI (possedeJardin? = oui) ALORS (typeDeChien = extérieur)

R18: SI (sportif? = oui) ET (possedeJardin? = non) ALORS (typeDeChien = extérieur)

R19: SI (tempsDeTravail = beaucoup) ET (nbEnfants > 3) ALORS (avoirDuTemps? = non)

R20: SI (tempsDeTravail = beaucoup) ET (nbEnfants <= 3) ALORS (avoirDuTemps? = moyen)

R21: SI (tempsDeTravail = moyen) ET (nbEnfants >= 3) ALORS (avoirDuTemps? = moyen)

R22: SI (tempsDeTravail = moyen) ET (nbEnfants < 3) ALORS (avoirDuTemps? = oui)

R23: SI (tempsDeTravail = faible) ET (nbEnfants > 3) ALORS (avoirDuTemps? = moyen)

R24: SI (tempsDeTravail = faible) ET (nbEnfants <= 3) ALORS (avoirDuTemps? = oui)

R25: SI (avoirDuTemps? = non) ALORS (supporteSolitude = oui)

R26: SI (avoirDuTemps? = moyen)) (supporteSolitude = moyen)

R27: SI (avoirDuTemps? = oui)) (supporteSolitude = non)

R28: SI (> nbEnfants 3) ALORS (caractereIndividu = autoritaire)

R29: SI (caracteristiqueIndividu = aveugle) ALORS (caracteristiqueChien = guide)

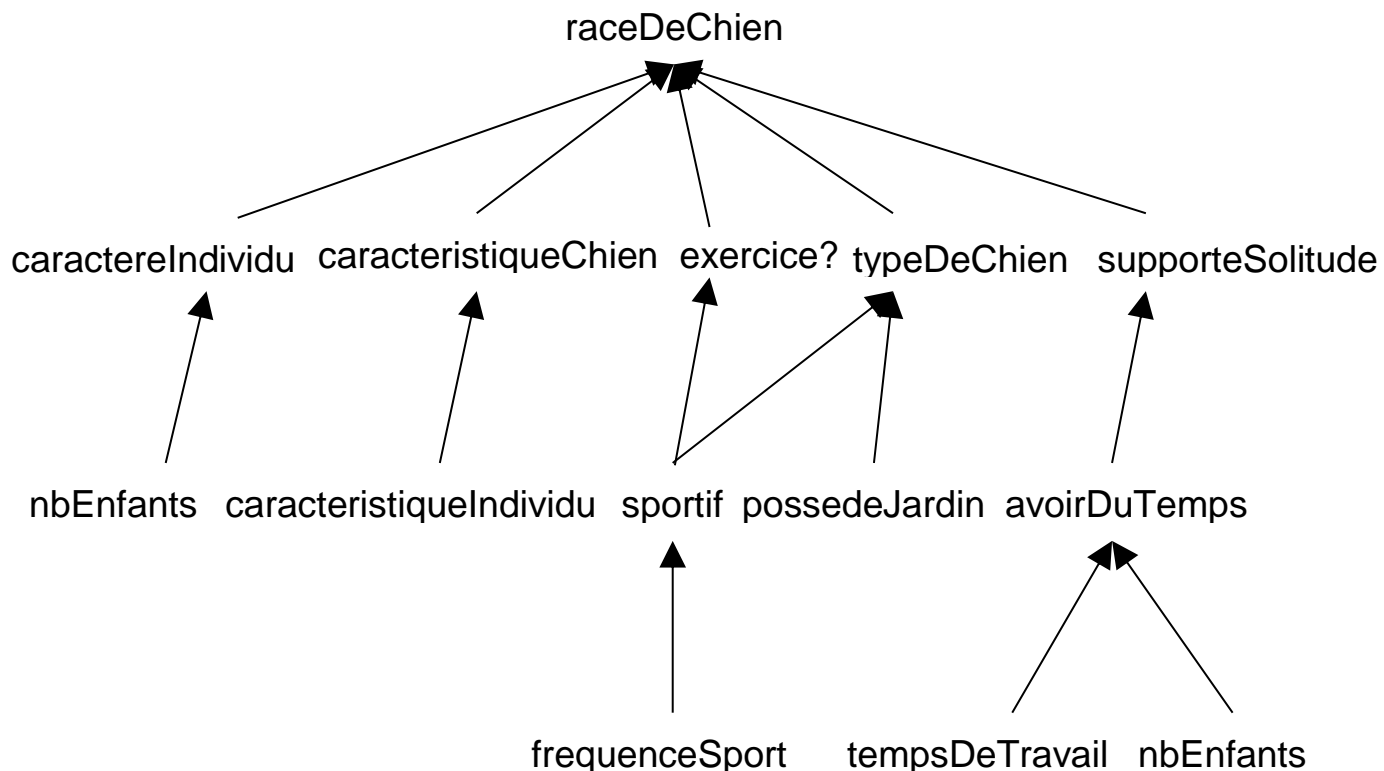
R30: SI (caracteristiqueIndividu = age) ALORS (caracteristiqueChien = compagnie)

R31: SI (caracteristiqueIndividu = age) ALORS (caracteristiqueChien = guide)

R32: SI (caracteristiqueIndividu = aucune) ALORS (caracteristiqueChien = compagnie)

R33: SI (caracteristiqueIndividu = aucune) ALORS (caracteristiqueChien = guide)

### Arbre de déduction:



Une race de chien peut être déterminée par le caractère de l'individu, s'il est autoritaire ou non, par une caractéristique du chien, si c'est un chien de compagnie ou un chien guide, le type de chien, s'il a besoin d'être en extérieur ou non, si le chien a besoin d'exercice ou non et si le chien supporte la solitude.

### Jeux d'essais:

Si on part des caractéristiques suivantes pour un individu: **nbEnfants=0**, **frequenceSport=1fois/semaine**, **possedeJardin?=non**, **tempsDeTravail=moyen** et **caracteristiqueIndividu=aucune**

On obtient les races de chien suivantes: Labrador, Pekinois, Carlin, BichonFrise et Boberman.

Les règles appliquées ont été à la suite: R12, R15, R22, R27, R32, R1, R3, R4, R5, R33 et R9.

Si on prend l'exemple suivant: **nbEnfants=4**, **frequenceSport=2fois/semaine**, **possedeJardin?=oui**, **tempsDeTravail=moyen**, **caracteristiqueIndividu=aucune** et **caractereIndividu=autoritaire**

On obtient les races de chien suivantes: BergerAllemand, Leonberger, Chihuahua, MalamuteDeLAlaska, Pekinois, Carlin et Beauceron.

Les règles appliquées ont été à la suite: R11, R14, R17, R21, R26, R2, R28, R32, R4, R5, R6, R7, R8, R33 et R10.

### Question 3

#### Représentation des faits, des règles et des bases:

Nous avons choisi de représenter nos règles par des a-listes de la forme:

(Etiquette Premises Conclusion)

Etiquette est un identifiant de la forme RX, X compris entre 1 et le nombre de règles existantes, permettant de différencier les différentes règles et de leur attribuer un identifiant unique. Cela permettra de manipuler les règles par l'intermédiaire de leur identifiant.

Premises est une liste des prémisses des règles de la forme ((Premisse1) (Premisse2) ...) où PremisseX est sous forme (opérateur objet valeur).

Conclusion est de la forme (opérateur objet valeur).

Notre base de règle est une liste de toutes nos règles. Elle contient d'abord les règles permettant d'arriver aux races de chien, puis les règles permettant de faire d'autres déduction.

Notre base de fait est également une liste de tous nos faits, mais les faits sont représentés comme suit: (objet valeur). On a considéré que tous les couples objet-valeur était lié par l'opérateur égal, c'est pourquoi il n'est pas représenté dans les couples de fait.

La base de fait est rempli directement par l'utilisateur grâce à la fonction createBF. Il est indiqué entre parenthèse quel type de donné est attendu. Nous n'avons pas laissé à l'utilisateur la possibilité de répondre "Je ne sais pas" à une question. En effet, les questions posées ne sont pas compliquées ou ne nécessitent pas de réflexion, elles concernent directement la vie de l'individu.

#### Présentation du moteur d'inférence:

Nous avons choisi d'implémenter un moteur d'inférence en chaînage avant en profondeur d'abord. En effet, notre but est de connaître toutes les races de chien dont peut s'occuper un individu. En chaînage arrière, il aurait fallu partir de toutes les races de chien et regarder si la base de fait permettrait d'en déduire individuellement toutes ces races de chien. En chaînage avant le procédé est plus simple, on part de la base de fait et on en déduit les races correspondantes.

Voici le moteur implémenté:

(defun moteur ())

(let\* ((a (createBF)) (candidates (reglesCandidates)) (regle (car candidates)) (liste NIL) (conclusion (getConclusion regle)))

(loop

(if candidates

(progn

(setq conclusion (getConclusion regle))

(declencher regle)

(if (equal 'raceDeChien (cadar conclusion))

(push (cadar \*BF\*) liste)

)

La base de fait est rempli avec la fonction `createBF`, qui pose des questions à l'utilisateur pour construire les couples (objet valeur) correspondant à ses réponses.

Une fois une règle appliquée, on vérifie si c'est une règle permettant de déduire une race de chien. Si c'est le cas on ajoute la race de chien à une liste.

### Test du moteur d'inférence:

Plusieurs tests ont été effectués pour tester l'efficacité du moteur:

Cette étudiante pourrait s'occuper d'un Labrador, un Pékinois, un Carlin, un Bichon Frisé et un Doberman.

```
CG-USER(48): *BF*
NIL
CG-USER(49): (moteur)
A quel frequence pratiquez-vous du sport par semaine ? (en chiffre)
1
Combien d'enfants avez-vous ? (en chiffre)
0
Possédez-vous un jardin ? (oui ou non)
non
Travaillez-vous beaucoup (oui, moyen ou non)
moyen
Avez-vous une caractéristique particulière ? (aveugle, age ou aucune)
aucune
Avez-vous un caractère particulier ? (autoritaire ou aucun)
aucun
("Labrador" "Pekinois" "Carlin" "BichonFrise" "Boberman")
```

Prenons maintenant l'exemple d'un adulte qui ferait du sport deux fois par semaine, qui aurait quatre enfant, vivrait dans un pavillon avec un jardin, travaillant moyennement et étant autoritaire. Cette personne pourrait s'occuper d'un Berger Allemand, un Leonberger, un Chihuahua, un Malamute de l'Alaska, un Pékinois, un Carlin et Beauceron. Les différences majeures par rapport à l'étudiant sont qu'il a moins de temps à accorder à son animal du fait qu'il ait quatre enfant, mais il a un jardin donc il pourrait avoir un chien ayant besoin d'être à l'extérieur et il est autoritaire, ce qui lui permet entre autre de prendre un berger allemand.

```
CG-USER(44): *BF*
NIL
CG-USER(45): (moteur)
A quel frequence pratiquez-vous du sport par semaine ? (en chiffre)
2
Combien d'enfants avez-vous ? (en chiffre)
4
Possédez-vous un jardin ? (oui ou non)
oui
Travaillez-vous beaucoup (oui, moyen ou non)
moyen
Avez-vous une caractéristique particulière ? (aveugle, age ou aucune)
aucune
Avez-vous un caractère particulier ? (autoritaire ou aucun)
autoritaire
("BergerAllemand" "Leonberger" "Chihuahua" "MalamuteDeLAlaska"
"Pekinois" "Carlin" "Beauceron")
-- .....
```

Si on prend l'exemple d'une personne âgée ne faisait pas de sport, n'ayant pas d'enfant, ne travaillant pas beaucoup et ayant une maison avec jardin, elle pourrait s'occuper d'un Pékinois, d'un Carlin ou d'un Bichon frisé. Par rapport à l'étudiant et à l'adulte, cette personne âgée ne fait pas de sport, c'est pourquoi il y a moins de races de chien correspondantes.

```
CG-USER(37): *BF*
NIL
CG-USER(38): (moteur)
A quel frequence pratiquez-vous du sport par semaine ? (en chiffre)
0
Combien d'enfants avez-vous ? (en chiffre)
0
Possédez-vous un jardin ? (oui ou non)
oui
Travaillez-vous beaucoup (oui, moyen ou non)
non
Avez-vous une caractéristique particulière ? (aveugle, age ou aucune)
age
Avez-vous un caractère particulier ? (autoritaire ou aucun)
aucun
("Pekinois" "Carlin" "BichonFrise")
```

Enfin prenons le cas d'une personne aveugle qui ne fait pas de sport, qui a deux enfants, une maison avec un jardin et qui travaille moyennement. Aucune race de chien présente dans la base de connaissance ne correspond à cette personne.

```
CG-USER(53): *BF*
NIL
CG-USER(54): (moteur)
A quel frequence pratiquez-vous du sport par semaine ? (en chiffre)
0
Combien d'enfants avez-vous ? (en chiffre)
2
Possédez-vous un jardin ? (oui ou non)
oui
Travaillez-vous beaucoup (oui, moyen ou non)
moyen
Avez-vous une caractéristique particulière ? (aveugle, age ou aucune)
aveugle
Avez-vous un caractère particulier ? (autoritaire ou aucun)
aucun
"Aucune race de chien ne vous correspond."
```

### Démonstration du bon fonctionnement des différentes fonctions:

Tous les exemples d'application des fonctions de services seront fait avec cette base de fait de départ:

```
(setq *BF* '((nbEnfants 0) (frequenceSport 1) (possedeJardin? "non") (tempsDeTravail
"moyen") (caracteristiqueIndividu "aucune"))
)
```

- **getRegle (etiquette)** renvoie une règle à partir de son étiquette

```
CG-USER(192): (getRegle 'R1)
(R1
 ((EQUAL CARACTERISTIQUECHIEN "compagnie") (>= SPORTIF? 2)
  (EQUAL TYPEDECHIEN "interieur") (= SUPPORTESOLITUDE 1))
 (EQUAL RACEDECHIEN "Boberman"))
```

- **getPremisses (etiquette)** renvoie une liste des prémisses d'un règle à partir de son étiquette

```
CG-USER(193): (getPremisses 'R1)
((EQUAL CARACTERISTIQUECHIEN "compagnie") (>= SPORTIF? 2)
 (EQUAL TYPEDECHIEN "interieur") (= SUPPORTESOLITUDE 1))
```

- **getConclusion (etiquette)** renvoie une liste de la conclusion d'un règle à partir de son étiquette

```
CG-USER(194): (getConclusion 'R1)
((EQUAL RACEDECHIEN "Boberman"))
```

- **supprimeRegle (etiquette)** supprime une règle de la base de règles à partir de son étiquette



```
CG-USER(195): (supprimeRegle 'R1)
T
CG-USER(196): *BR*
((R2
  ((EQUAL CARACTERISTIQUECHIEN "gardien") (>= SPORTIF? 2)
   (EQUAL POSSEDEJARDIN? "oui") (<= SUPPORTESOLITUDE 2))
  (EQUAL RACEDECHIEN "Beauceron"))
(R3
```

- **getValeur (objet)** renvoie la valeur d'un objet dans la base de faits

```
CG-USER(197): (getValeur 'nbEnfants)
0
```

- **setValeur (objet valeur)** permet de modifier la valeur d'un objet existant dans la base de faits par la valeur passé en paramètre

```
CG-USER(198): (setValeur 'nbEnfants 5)
5
CG-USER(199): *BF*
((NBENFANTS 5) (FREQUENCESPORT 1) (POSSEDEJARDIN? "non")
 (TEMPSDETRAVAIL "moyen") (CARACTERISTIQUEINDIVIDU "aucune"))
```

- **ajoutDansBF (couple)** permet d'ajouter un couple (objet valeur) à la base de faits. Si le couple existe déjà on modifie uniquement sa valeur

```
CG-USER(200): (ajoutDansBF '(sportif? 2))
T
CG-USER(201): *BF*
((SPORTIF? 2) (NBENFANTS 5) (FREQUENCESPORT 1) (POSSEDEJARDIN? "non")
 (TEMPSDETRAVAIL "moyen") (CARACTERISTIQUEINDIVIDU "aucune"))
```

- **createBF ()** permet de questionner l'utilisateur et d'ajouter les couples correspondant à ses réponses à la base de faits

```
CG-USER(101): (setq *BF* '())
NIL
CG-USER(102): (createBF)
A quel frequence pratiquez-vous du sport par semaine ? (en chiffre)
1
Combien d'enfants avez-vous ? (en chiffre)
0
Possédez-vous un jardin ? (oui ou non)
non
Travaillez-vous beaucoup (oui, moyen ou non)
moyen
Avez-vous une caractéristique particulière ? (aveugle, age ou aucune)
aucune
Avez-vous un caractère particulier ? (autoritaire ou aucun)
aucun
T
CG-USER(103): *BF*
((CARACTEREINDIVIDU "aucun") (CARACTERISTIQUEINDIVIDU "aucune")
 (TEMPSDETRAVAIL "moyen") (POSSEDEJARDIN? "non") (NBENFANTS 0)
 (FREQUENCESPORT 1))
```

- **checkPremisse (p)** permet de vérifier si une prémisse est vrai par rapport à la base de faits

```
CG-USER(205): (checkpremise '(>= SPORTIF? 2))
T
CG-USER(206): (checkpremise '(= nbEnfants 6))
NIL
```

- **declenchable? (regle)** permet de vérifier si une règle est déclenchable, c'est-à-dire si elle peut être utilisée par le moteur d'inférence

```
CG-USER(216): (getRegle 'R1)
(R1
 ((EQUAL CARACTERISTIQUECHIEN "compagnie") (>= SPORTIF? 2)
 (EQUAL TYPEDECHIEN "interieur") (= SUPPORTESOLITUDE 1))
 (EQUAL RACEDECHIEN "Boberman"))
CG-USER(217): *BF*
((SPORTIF? 2) (NBENFANTS 5) (FREQUENCESPORT 1) (POSSEDEJARDIN? "non")
 (TEMPSDETRAVAIL "moyen") (CARACTERISTIQUEINDIVIDU "aucune"))
CG-USER(218): (declenchable? 'R1)
NIL
CG-USER(219): (declenchable? 'R9)
T
```

- **reglesCandidates ()** permet de récupérer la liste des règles candidates, qui peuvent être déclencher

```
CG-USER(220): (reglesCandidates)
(R9 R13 R17 R24 R28 R29)
```

- **declencher (regle)** permet de déclencher une règle

```
CG-USER(224): *BF*
((SPORTIF? 2) (NBENFANTS 5) (FREQUENCESPORT 1) (POSSEDEJARDIN? "non")
 (TEMPSDETRAVAIL "moyen") (CARACTERISTIQUEINDIVIDU "aucune"))
CG-USER(225): (getRegle 'R17)
(R17 ((EQUAL TEMPSDETRAVAIL "moyen") (>= NBENFANTS 3))
 (EQUAL AVOIRDUTEMPS? "moyen"))
CG-USER(226): (declencher 'R17)
T
CG-USER(227): *BF*
((AVOIRDUTEMPS? "moyen") (SPORTIF? 2) (NBENFANTS 5)
 (FREQUENCESPORT 1) (POSSEDEJARDIN? "non") (TEMPSDETRAVAIL "moyen")
 (CARACTERISTIQUEINDIVIDU "aucune"))
```

## Conclusion

Ce projet nous a permis de réaliser un système expert du début à la fin, de la recherche d'une idée à la réalisation du moteur.

Plusieurs améliorations pourraient être apportées:

- Notre base de règles pourrait être complétée avec d'autres races de chien. Mais le nombre de chien ayant des caractéristiques identiques seraient alors très important compte tenu de notre arbre de déduction actuel. Il faudrait par conséquent rechercher d'autres critères discriminants pour les différentes races de chien et donc ajouter de nouvelles questions pour l'utilisateur
- Un moteur en chaînage arrière pourrait être mis en place pour déterminer cette fois pour une race de chien précise si elle correspondrait à un individu donné