

BAZE DE CUNOȘTINȚE

REȚELE SEMANTICE

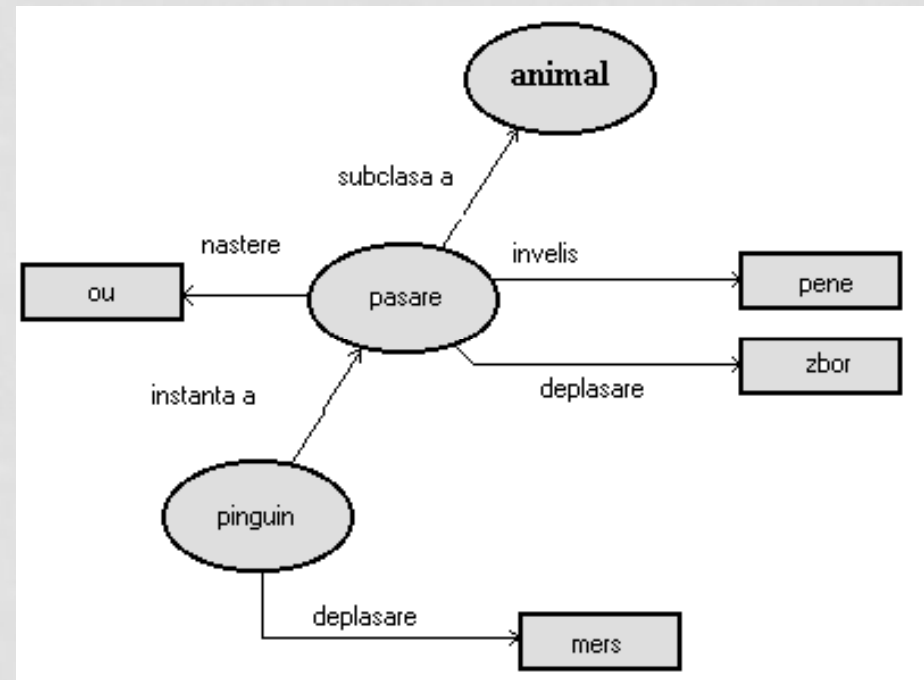
M C
I O
H L
A H
E O
L N
A

DEFINIȚII

Rețeaua semantică este o metodă de tip graf pentru reprezentarea cunoștințelor.

O rețea semantică constă din entități și relații între acestea.

Ea se reprezintă sub forma unui graf orientat ale cărui arce leagă vârfuri etichetate.

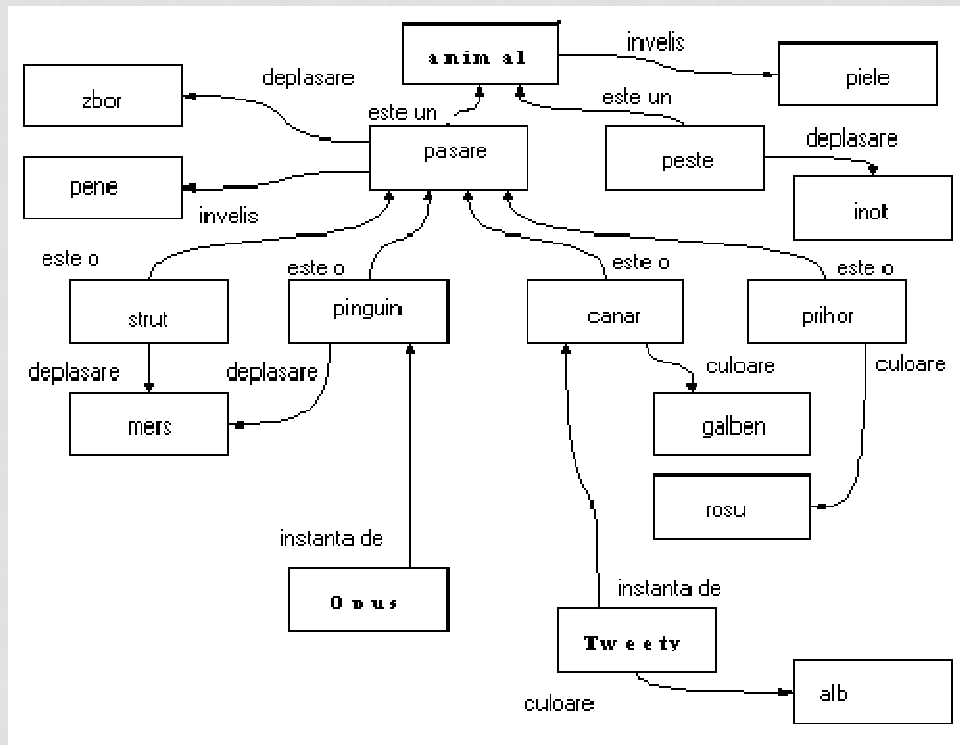


DEDUCȚII AUTOMATE ÎNTR-O REȚEA SEMANTICĂ

Rețeaua semantică permite efectuarea de *deducții automate*. Astfel de raționamente se bazează pe reguli de forma:

- Dacă **orice** X este Y și **orice** Y este Z atunci **orice** X este Z .
- Dacă X este Y și **orice** Y este Z atunci X este Z .
- Dacă **orice** X este Y și Y este Z atunci **orice** X este Z .
- Dacă X este Y și Y este Z atunci X este Z .

EXEMPLU



Se observă că relația de tip **ISA/ este_un** realizează legătura dintre o clasă de obiecte și altă clasă, mai generală.

Altfel, relația ISA face legătura dintre o **instanță** a unei clase și clasa însăși.

MOȘTENIREA PROPRIETĂȚILOR

Principiul tipic de inferență încorporat într-o rețea semantică este **moștenirea proprietăților**. Faptele sunt moștenite prin intermediul relației ISA.

Implementarea în Prolog a moștenirii

```
inherit:- isa(X, SuperX), relation(SuperX, name, Y),  
assert(relation(X, name, Y)), inherit.  
inherit.
```

COMPUNEREA RELAȚIILOR

Compunerea a două relații φ_1 și φ_2 pe X este o nouă relație pe X , notată cu $\varphi_1 \cdot \varphi_2$ și definită prin:

$(x,y) \in \varphi_1 \cdot \varphi_2$ dacă $(\exists)z \in X: (x,z) \in \varphi_1$ și $(z,y) \in \varphi_2$

Compunerea relațiilor este asociativă:

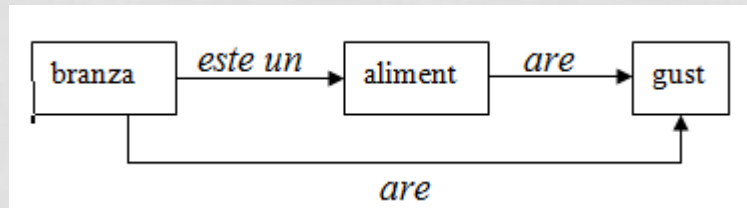
$$(\varphi_1 \cdot \varphi_2) \cdot \varphi_3 = \varphi_1 \cdot (\varphi_2 \cdot \varphi_3)$$

Compunerea a două relații poate fi relația vidă fără ca vreuna din relațiile care se compun să fie vidă.

Exemplu:

$$\varphi_1 = \{(x,y), (x,z)\} \text{ și } \varphi_2 = \{(x,v)\} \text{ atunci } \varphi_1 \cdot \varphi_2 = \emptyset$$

COMPUNEREA RELAȚIILOR



În baza acestei reprezentări obținem:

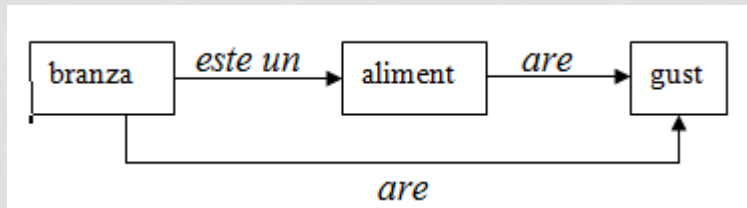
- obiecte: "branza", "aliment", "gust"
- relații binare:
- $\varphi_1 = \{("branza", "aliment")\}$
- $\varphi_2 = \{("aliment", "gust"), ("branza", "gust")\}$

Interpretarea relațiilor este următoarea:

$(x, y) \in \varphi_1$ înseamnă că x este un y

$(x, y) \in \varphi_2$ înseamnă că x are y

COMPUNEREA RELAȚIILOR



Relațiile binare într-o rețea semantică se pot compune dacă semnificatia permite.

De exemplu:

dacă $(x, y) \in \varphi_1$ și dacă $(y, z) \in \varphi_2$ atunci $(x, z) \in \varphi_2$.

DRUMURI CONFLUENTE

Daca avem in graful atasat retelei semantice doua drumuri care leaga pe **x** cu **z** si pe **y** cu **z**, atunci aceste drumuri se numesc **confluente**. Astfel intre obiectele **x** si **y** se stabileste o relatie noua care este legata de faptul ca cele doua drumuri conduc la acelasi obiect **z**.

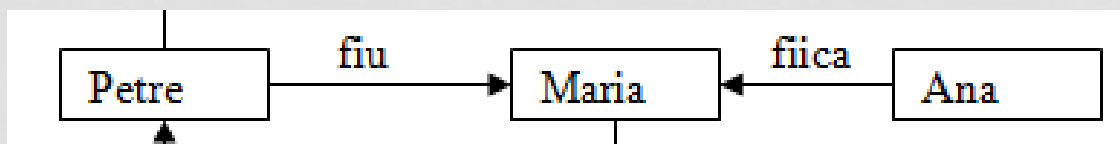
Astfel, dacă avem:

$$(x, y) \in \varphi_1$$

$$(z, y) \in \varphi_2$$

putem obține o nouă relație (x, z) sau (z, x) a cărei semnificație este dată de interpretarea lui relațiilor φ_1 și φ_2 .

DRUMURI CONFLUENTE



În baza acestei reprezentări obținem:

- obiecte: "Petre", "Maria", "Ana"
- $\varphi_1 = \{("Petre", "Maria")\}$, $\varphi_2 = \{("Ana", "Maria")\}$

Interpretarea relațiilor este următoarea:

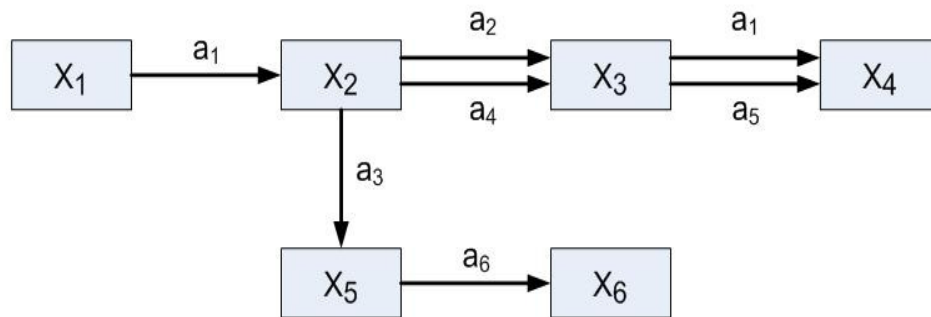
$(x, y) \in \varphi_1$ înseamnă că x este *fiul* lui y

$(x, y) \in \varphi_2$ înseamnă că x este *fiica* y

Din $(x, y) \in \varphi_1$ și $(z, y) \in \varphi_2$ obținem

- $(x, z) \in \varphi_3$ cu semnificația x este *fratele* lui z
- $(z, x) \in \varphi_4$ cu semnificația z este *sora* lui x

GRAF ETICHETAT



$a_1 \longrightarrow \{ (X_1, X_2), (X_3, X_4) \} = \rho_1$; $a_2, a_4 \longrightarrow \{ (X_2, X_3) \} = \rho_2$
 $a_3 \longrightarrow \{ (X_2, X_5) \} = \rho_3$; $a_5 \longrightarrow \{ (X_3, X_4) \} = \rho_4$
 $a_6 \longrightarrow \{ (X_5, X_6) \} = \rho_5$

- $S = \{ X_1, \dots, X_6 \}$
- $L_0 = \{ a_1, \dots, a_6 \}$
- $T_0 = \{ \rho_1, \dots, \rho_5 \}$
- $f_0 : L_0 \rightarrow T_0$ where
 $f_0(a_1) = \rho_1$; $f_0(a_2) = f_0(a_4) = \rho_2$
 $f_0(a_3) = \rho_3$; $f_0(a_5) = \rho_4$; $f_0(a_6) = \rho_5$

$G_0 = (S, L_0, T_0, f_0)$ este un graf etichetat

Algebra Peano generată de L_0 și σ este cea mai mică mulțime L astfel încât:

1) $L_0 \subseteq L$

2) $u, v \in L \Rightarrow \sigma(u, v) \in L$

ÎNCHIDEREA MULȚIMII T_0

Dacă $T_0 \subset 2^{X \times X}$ este o mulțime de relații binare pe mulțimea finită X , atunci se notează $C(T_0)$ cea mai mică mulțime (în sensul relației de incluziune) $A \subset 2^{X \times X}$ astfel încât:

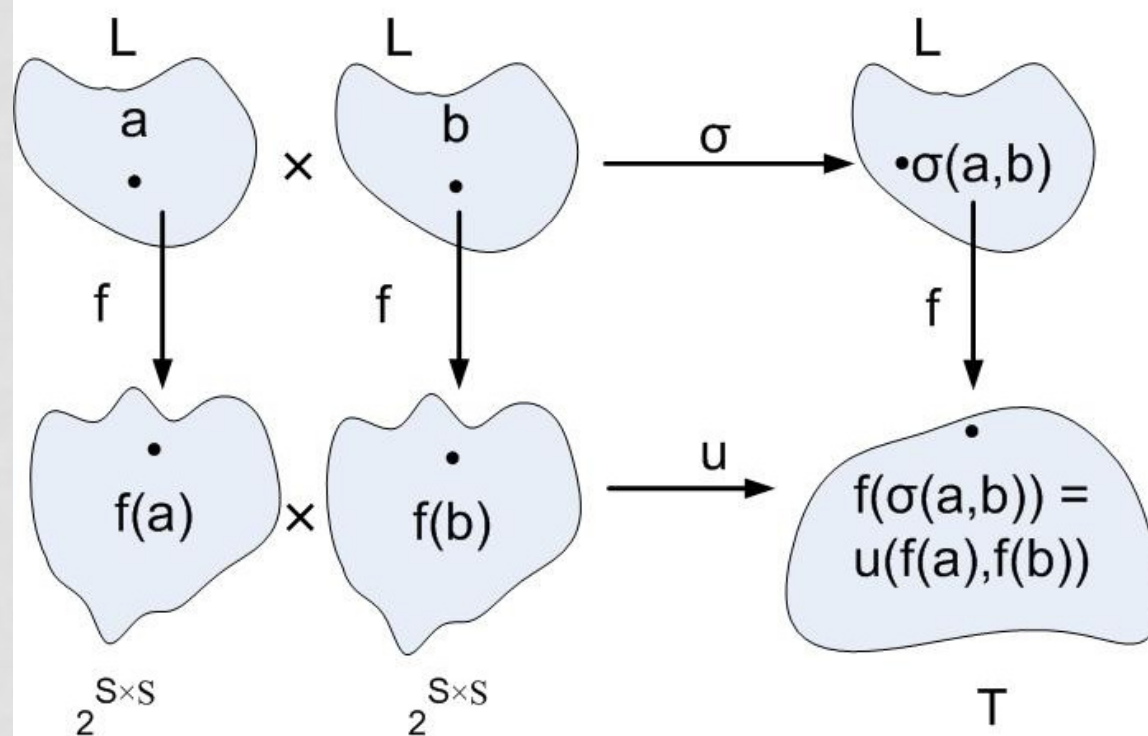
- $T_0 \subset A$
- dacă $\varphi_1, \varphi_2 \in A$ și $\varphi_1 \cdot \varphi_2 \neq \emptyset$ atunci $\varphi_1 \cdot \varphi_2 \in A$.

REȚEA SEMANTICĂ. FORMALISM

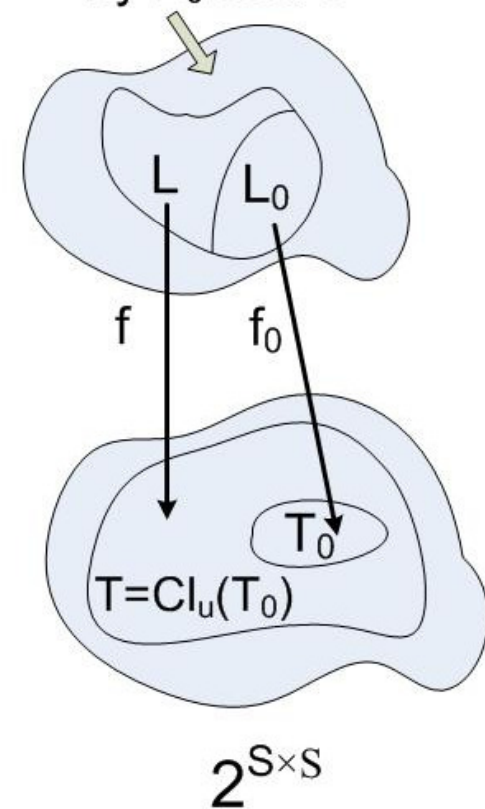
O **rețea semantică** este un tuplu

$$(G, T, L, f, u, g, S)$$

- $G=(X, L_0, T_0, f_0)$ este un graf etichetat
- $L_0 \subset L$ și $\sigma: L \times L \rightarrow L$ este o aplicație parțial definită
- $T_0 \subset T \subset C(T_0)$
- $f: L \rightarrow T$ este o extensie surjectivă a funcției f_0 a.î. dacă $(a,b) \in \text{dom}(\sigma)$ atunci $f(\sigma(a,b))=u(f(a),f(b))$
- S se numește **spațiu semantic**
- $g: X \times L \times X \rightarrow S$ se numește **aplicație semantică**



Peano algebra generated
by L_0 and σ



REȚEA SEMANTICĂ. FUNCȚIA ANS

În general, deoarece răspunsul la o interogare se dă în limbajul natural, spațiul semantic S este o mulțime de propoziții în acest limbaj. Intuitiv, elementul $g(x, e, y)$ specifică semantica faptului că e este eticheta pentru o anumită relație binară care conține perechea (x, y) .

Pentru o rețea semantică simplă, (G, T, L, f, u, g, S) , răspunsul la o interogare este dat de aplicația

$$\text{Ans: } X \times X \rightarrow 2^S \cup \{\text{unknown}\}$$

REȚEA SEMANTICĂ. FUNCȚIA ANS

$$\text{Ans}: X \times X \rightarrow 2^S \cup \{\text{unknown}\}$$

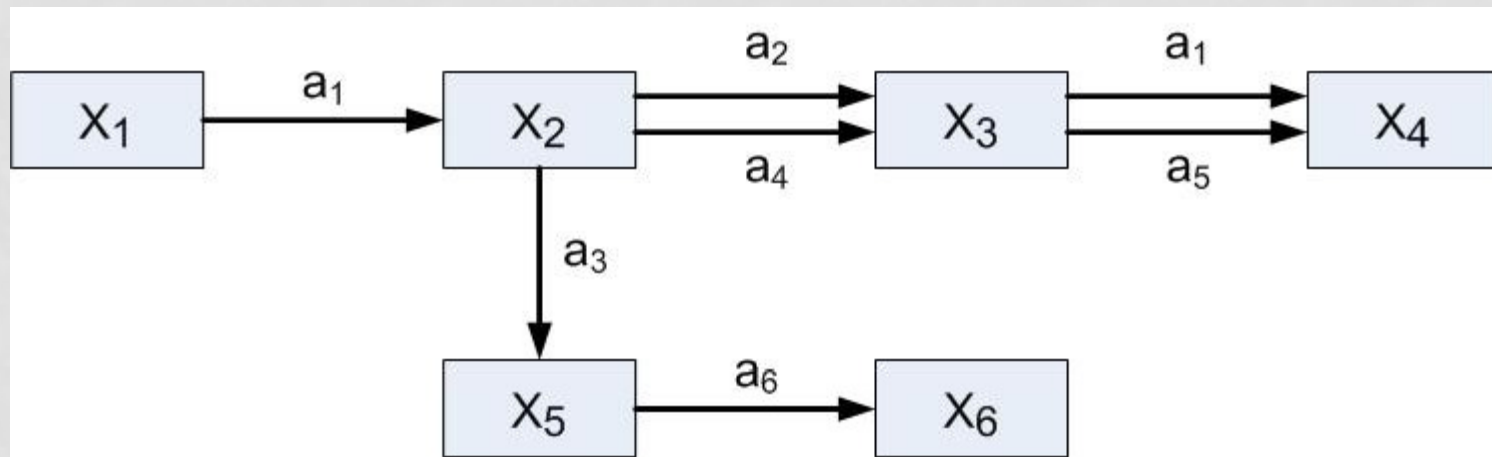
definită prin:

$$\text{Ans}(x, y) = \begin{cases} \text{unknown}, & \text{daca } \text{deduc}(x, y) = \emptyset \\ \text{deduc}(x, y), & \text{altfel} \end{cases}$$

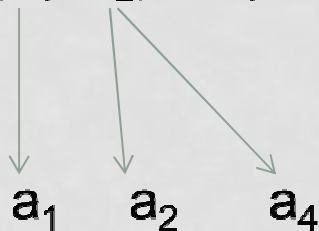
unde:

$$\text{deduc}(x, y) = \{g(x, u, y) \mid u \in L\}$$

EXEMPLIFICARE



- $f_0(a_1) = \rho_1$; $f_0(a_2) = f_0(a_4) = \rho_2$; $f_0(a_3) = \rho_2$; $f_0(a_5) = \rho_4$; $f_0(a_6) = \rho_5$
- Define $u(\rho_1, \rho_2) = \omega_1 = \{(x_1, x_3)\}$; $\text{dom}(u) = \{(\rho_1, \rho_2)\}$



- $f(\sigma(a_1, a_2)) = f(\sigma(a_1, a_4)) = \omega_1$; $T = T_0 \cup \{\omega_1\}$

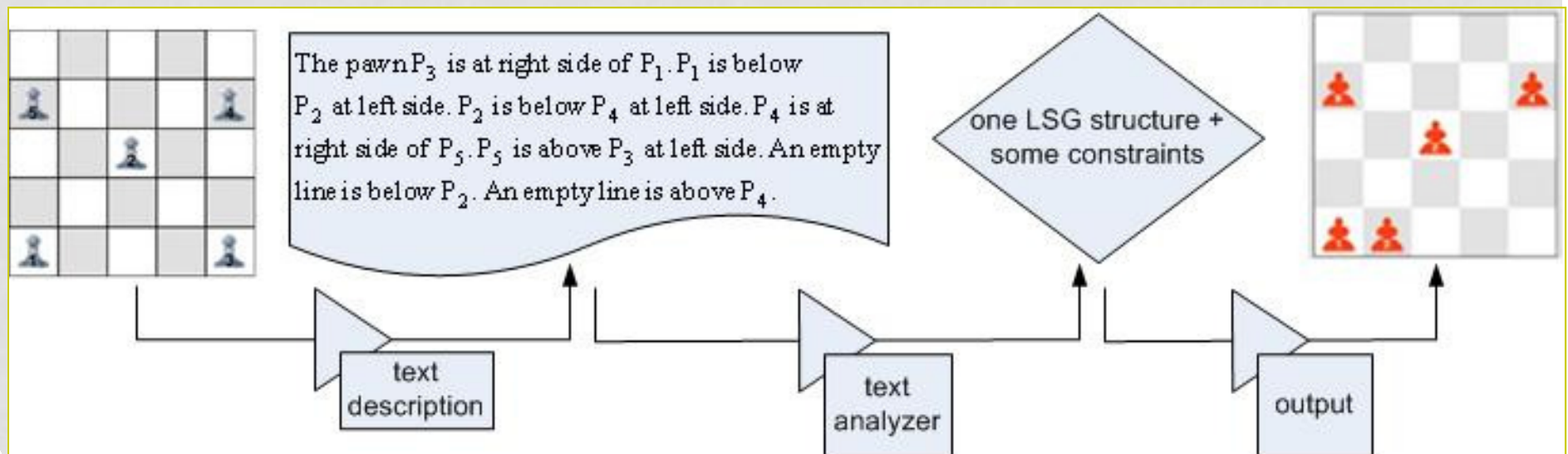
RĂȚIONAMENTE ÎN REȚELE SEMANTICE (I)

PROBLEM: To develop an *image synthesis method* by means of which we can *reconstruct a graphical image* from a *linguistic spatial description* given in a *natural language (English)*.

INPUT: text description

REPRESENTATION AND REASONING MECHANISM: semantic network

OUTPUT: grid representation of the described image



RĂȚIONAMENTE ÎN REȚELE SEMANTICE (II)

Applet Viewer: JIP.gui.SemNet.SemNetApplet.class

Applet

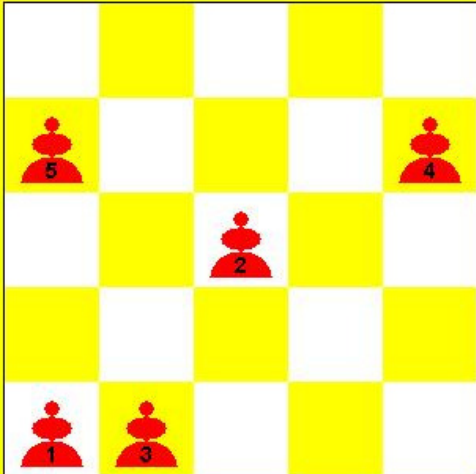
Load KP Draw Graph Draw Pawns Reset

Enter The Knowledge Piece

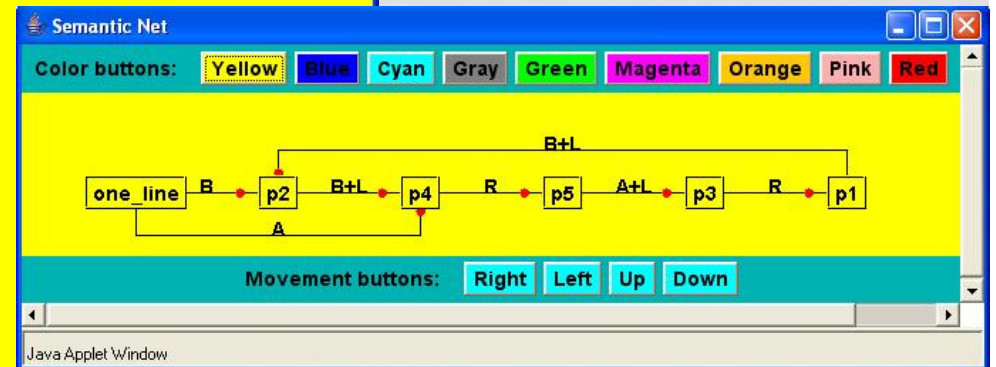
The pawn P3 is at right side of P1.
P1 is below P2 at left side.
P2 is below P4 at left side.
P4 is at right side of P5.
P5 is above P3 at left side.
One empty line is below P2.
One empty line is above P4.

The JIProlog Output Window

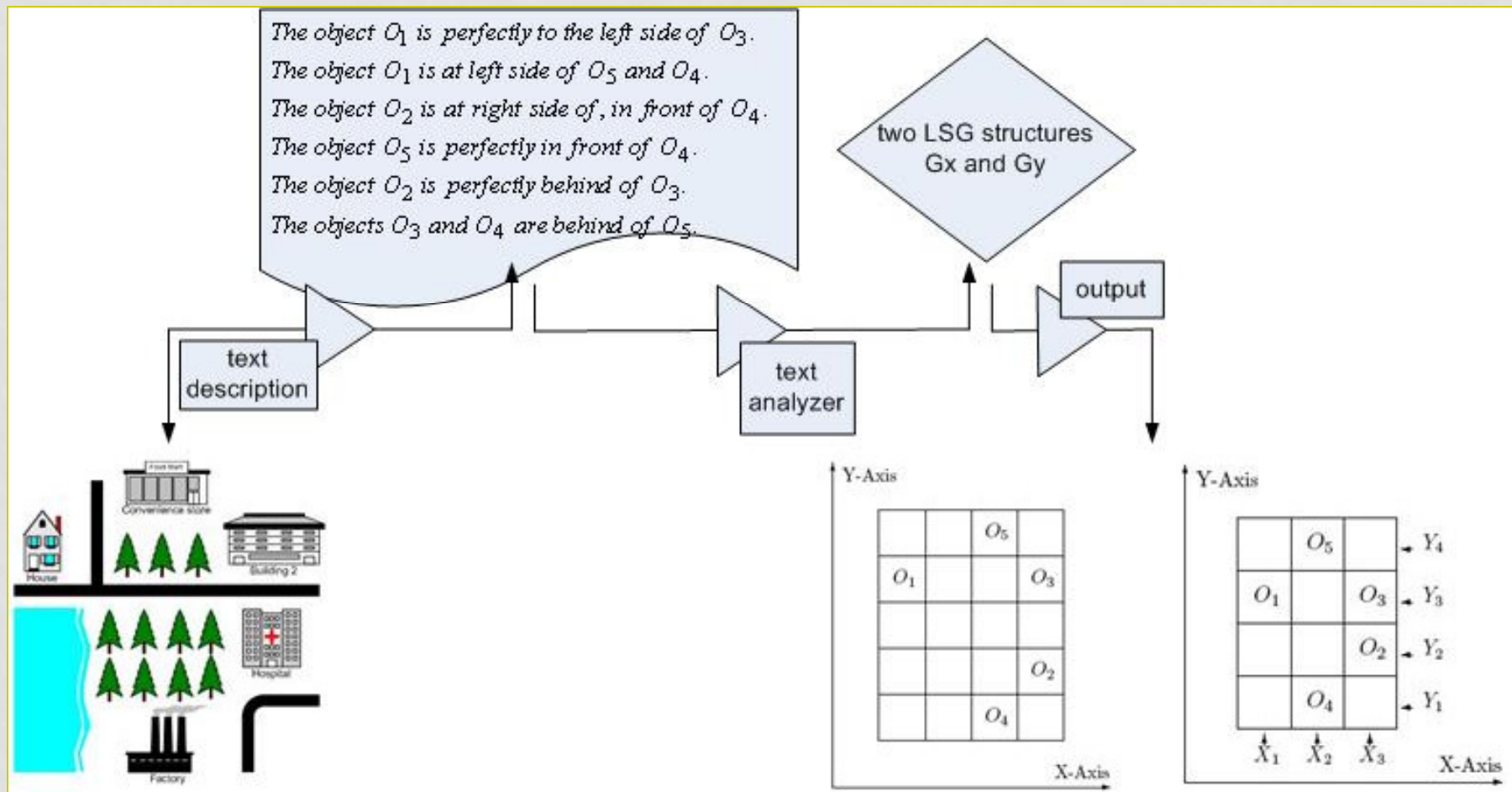
Solution notified: prop([p1, is, below, p2
Solution notified: prop([p2, is, below, p4
Solution notified: prop([p4, is, at, right, s
Solution notified: prop([p5, is, above, p3
Solution notified: prop([one, empty, line
Solution notified: prop([one, empty, line
Solution notified: finish
End notified



Applet started.



RAȚIONAMENTE ÎN REȚELE SEMANTICE (III)



Vă mulțumesc!