CSc 520
Principles of Programming Languages

44: Logic Programming — Executing Prolog

Christian Collberg
collberg+520@gmail.com

Department of Computer Science
University of Arizona

Copyright © 2008 Christian Collberg
Now that we know about matching, we can take a closer look at how Prolog tries to satisfy goals.

In general, to solve a goal

$$G = G_1, G_2, \ldots, G_m,$$

Prolog will first try to solve the sub-goal $G_1$.

It solves a sub-goal $G_1$ it will look for a rule

$$H_i : \leftarrow B_1, \ldots, B_n$$

in the database, such that $G_1$ and $H_i$ will match.

Any variable substitutions resulting from the match will be stored in a variable $\theta$. 

[2]
A new goal will be constructed by replacing $G_1$ with $B_1, \ldots, B_n$, yielding

$$G' = B_1, \ldots, B_n, G_2, \ldots, G_m.$$ 

If $n = 0$ the new goal will be shorter and we’ll be one step closer to a solution to $G$!

Any new variable bindings from $\theta$ are applied to the new goal, yielding $G''$.

We recursively try to find a solution to $G'''$. 
Executing Prolog...

```
FUNC Execute (G = G₁, G₂, ⋯, Gₘ; Result);
  IF Is_Empty(G) THEN Result := Yes
  ELSE
    Result := No;
    i := 1;
    WHILE Result=No & i ≤ NoOfClauses DO
      Clause := Hᵢ :- B₁, ⋯, Bₙ;
      IF Unify(G₁, Clause, θ) THEN
        G' := B₁, ⋯, Bₙ, G₂, ⋯, Gₘ;
        G'' := substitute(G', θ);
        Execute(G'', Result);
      ENDIF;
      i := i + 1;
    ENDDO
  ENDF;
ENDDIF
```
Executing Prolog

Goal
G1, G2, ..., Gm

Empty?

Scan database
Unify(Hi, G1)

Match?

Replace G1 by X1, ..., Xn
X1, ..., Xn, G2, ..., Gm
Substitute vars from θ
X1', ..., Xn', G2', ..., Gm'

No more Hi
fail

Yes
Succeed

θ = {⋯}

Hi := X1, ..., Xn

No

Yes

Database
(1) H1 := A1, ..., An
(2) H2 := B1, ..., Bn
(3) H3 := C1, ..., Cn

......
% From the Northern Exposure FAQ
% friend(of, kind(name, regular)).
friend(maggie, person(eve, yes)).
friend(maggie, moose(morty, yes)).
friend(maggie, person(harry, no)).
friend(maggie, person(bruce, no)).
friend(maggie, person(glenn, no)).
friend(maggie, person(dave, no)).
friend(maggie, person(rick, no)).
friend(maggie, person(mike, yes)).
friend(maggie, person(joel, yes)).
Northern Exposure Example...

cause_of_death(morty, copper_deficiency).
cause_of_death(harry, potato_salad).
cause_of_death(bruce, fishing_accident).
cause_of_death(glenn, missile).
cause_of_death(dave, hypothermia).
cause_of_death(rick, hit_by_satellite).
cause_of_death(mike, none_yet).
cause_of_death(joel, none_yet).

male(morty).  male(harry).  male(bruce).
male(glenn).  male(dave).  male(rick).
male(mike).  male(joel).  female(eve).
Northern Exposure Example...

alive(X) :- cause_of_death(X, none_yet).
pastime(eve, hypochondria).
pastime(mike, hypochondria).
pastime(X, golf) :- job(X, doctor).

job(mike, lawyer). job(adam, chef).
job(maggie, pilot). job(joel, doctor).

?- friend(maggie, person(B, yes)),
   male(B),
   alive(B),
   pastime(B, golf).
Northern Exposure Example...

friend(maggie, p(B, yes)).
male(B), alive(B),
pastime(B, golf).

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
alive(X) :- cause_od(X, none).
male(mike). male(joel).
female(eve).
pastime(eve, hypocondriac).
pastime(mike, hypocondriac).
pastime(X, golf) :- job(X, doctor).
job(adam, chef).
job(joel, doctor).

match?

Unify(Hi, G1)

Scan database

Empty? Yes

No

No more Hi fail

Hi :- X1, ..., Xn
\[ \theta = \{ B = eve \} \]

Replace G1 by <empty>
Substitute vars from \( \theta \)

male(eve), alive(eve)
pastime(eve, golf).
Northern Exposure Example...

G1

male(eve), alive(eve), pastime(eve, golf).

friend(m,p(eve,yes)).
friend(m,m(morty, yes)).
friend(m,p(harry, no)).
friend(m,p(mike, yes)).
friend(m,p(joel, yes)).
cause_od(mike,none).
cause_od(joel,none).
alive(X):−cause_od(X, none).
male(mike).male(joel).
female(eve).
pastime(eve, hypocondriac).
pastime(mike, hypocondriac).
pastime(X,golf):−job(X,doctor).
job(adam, chef).
job(joel, doctor).

Scan database
Unify(Hi, G1)

Empty?

Yes

Succeed

No

Match?

Yes

No more Hi

fail

(1)
Northern Exposure Example...

friend(maggie, p(B, yes)).
male(B), alive(B),
pastime(B, golf).

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
alive(X) :- cause_od(X, none).
male(mike). male(joel).
female(eve).
pastime(eve, hypocondriac).
pastime(mike, hypocondriac).
pastime(X, golf) :- job(X, doctor).
job(adam, chef).
job(joel, doctor).

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).

---

Hi :− X1, ..., Xn
θ = {B=mike}

Replace G1 by <empty>
Substitute vars from θ

male(mike), alive(mike),
pastime(mike, golf).
Hi (1)

male(mike), alive(mike),
pastime(mike, golf).

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
avive(X) :- cause_od(X, none).

job(adam, chef).
job(joel, doctor).

male(mike), alive(mike), pastime(mike, golf).

Replace G1 by <empty>
Northern Exposure Example...

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
alive(X):- cause_od(X, none).
job(adam, chef).
job(joel, doctor).

G1

alive(mike),
pastime(mike, golf).

Scan database
Unify(Hi, G1)

Hi :- X1, ..., Xn
θ = {X=mike}

Replace G1 by
<cause_od(X, none)>

cause_od(mike, none),
pastime(mike, golf).

Hi

Match?

Yes

Succeed

No

Yes

No more Hi
fail

Empty?

Yes

No
Northern Exposure Example...

friend(m,p(eve,yes)).
friend(m,m(morty,yes)).
friend(m,p(harry,no)).
friend(m,p(mike,yes)).
friend(m,p(joel,yes)).
cause_od(mike,none).
cause_od(joel,none).
aalive(X):-cause_od(X, none).
male(mike). male(joel).
female(eve).
pastime(eve, hypoc).
pastime(mike, hypoc).
pastime(X,golf):-job(X,doctor).
job(adam, chef).
job(joel, doctor).

cause_od(mike, none), pastime(mike, golf).

Hi :− X1, ..., Xn
θ = {}
Northern Exposure Example...

friend(m,p(eve, yes)). (1)
friend(m,m(morty, yes)). (4)
friend(m,p(harry, no)).
friend(m,p(mike, yes)).
friend(m,p(joel, yes)). (2)
cause_od(mike, none).
cause_od(joel, none). (3)
alive(X):-cause_od(X, none).
male(mike). male(joel).
female(eve).
pastime(eve, hypoc).
pastime(mike, hypoc).
pastime(X,golf):-job(X,doctor).
job(adam, chef).
job(joel, doctor).

520 — Spring 2008 — 44 [16]
Northern Exposure Example...

- We skip a step here.
- \texttt{pastime(mike, golf)} unifies with \texttt{pastime(X, golf) :- job(X, doctor)}.

- However, \texttt{job(mike, doctor)} fails, and we backtrack all the way up to the original query.
Northern Exposure Example...

friend(maggie, p(B, yes)).
male(B), alive(B),
pastime(B, golf).

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
alive(X):-cause_od(X, none).

job(adam, chef).
job(joel, doctor).
pastime(eve, hypocondriac).
pastime(mike, hypocondriac).
pastime(X, golf):-job(X, doctor).

G1

Hi

No

Scan database
Unify(Hi, G1)

Match?

Yes

Replace G1 by <empty>
Substitute vars from \( \theta \)

Hi :- X_1, \ldots, X_n
\( \theta = \{ B=joel \} \)

male(joel), alive(joel),
pastime(joel, golf).
Northern Exposure Example...

Scan database
Unify(Hi, G1)

Hi := X1, ..., Xn
θ = {X=joel}

Replace G1 by
job(X,doctor)
job(joel,doctor)
Northern Exposure Example...

job(joel, doctor) → G1

friend(m, p(eve, yes)).
friend(m, m(morty, yes)).
friend(m, p(harry, no)).
friend(m, p(mike, yes)).
friend(m, p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
alive(X) :- cause_od(X, none).
male(mike). male(joel).
female(eve).
pastime(eve, hypoc).
pastime(mike, hypoc).
pastime(X, golf) :- job(X, doctor).
job(adam, chef).
job(joel, doctor).

Hi := X1, ..., Xn
θ = {}

Match?

Empty?

Scan database
Unify(Hi, G1)

Yes

No

Succeed

No more Hi → fail

Replace G1 by <empty>
Northern Exposure Example...

friend(m,p(eve, yes)).
friend(m,m(morty, yes)).
friend(m,p(harry, no)).
friend(m,p(mike, yes)).
friend(m,p(joel, yes)).
cause_od(mike, none).
cause_od(joel, none).
alive(X):-cause_od(X, none).
male(mike). male(joel).
female(eve).
pastime(eve, hypoc).
pastime(mike, hypoc).
pastime(X,golf):-job(X,doctor).
job(adam, chef).
job(joel, doctor).
Readings and References

- Read Clocksin-Mellish, Section 4.1.
- See [http://www.moosefest.org](http://www.moosefest.org) for information about the annual Moosefest.
- See [http://members.lycos.co.uk/janineturner/engl/index.html](http://members.lycos.co.uk/janineturner/engl/index.html) for pictures of Janine Turner, who plays Maggie.
- See [http://home.comcast.net/~mcnotes/mcnotes.html](http://home.comcast.net/~mcnotes/mcnotes.html) for show transcripts.
A term is either a
- a constant (an atom or integer)
- a variable
- a structure

Two terms *match* if
- there exists a variable substitution $\theta$ which makes the terms identical.

Once a variable becomes instantiated, it stays instantiated.

Backtracking *undoes* variable instantiations.

Prolog searches the database sequentially (from top to bottom) until a matching clause is found.