Copyright ©1998 Timothy Howard Merrett
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation in a prominent place. Copyright for components of this work owned by others than T. H. Merrett must be honoured. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or fee. Request permission to republish from: T. H. Merrett, School of Computer Science, McGill University, fax 514 398 3883.

The author gratefully acknowledges support from the taxpayers of Québec and of Canada who have paid his salary and research grants while this work was developed at McGill University, and from his students (who built the implementations and investigated the data structures and algorithms) and their funding agencies.

T. H. Merrett
©98/10
Recursion, Knowledge, and Logic

- Hierarchies
  - Ancestor
  - Part-Of

- Syllogisms
  - ISA
  - LAPS

- Horn Clauses
Semantic Nets

Person — — — Name, Age, Address

Student — — — StudID

Sue

Staff — — — Prof

Tomasz

Employee — — — EmpID

Course — — — Time, Place

T

a

k

es

isA

hasA

in

other

Compute the closure of this under inheritance (isA).

T. H. Merrett
Semantic Nets

SemNet(First | Rel | Second)

Person hasA Name
Person hasA Age
Person hasA Address
Employee isA Person
Employee hasA EmpID
Student isA Person
Student hasA StudID
Student takes Course
Sue in Student
Prof isA Employee
Prof teaches Course
Staff isA Employee
Tomasz in Prof
Course hasA Time
Course hasA Place
617 in Course
Semantic Nets

Closure should give attribute inheritance, etc:

```
SemNet*(First  Rel  Second)
    :  :  :
    Prof  isA  Person
    Student  hasA  Name
    :  :  :
    Employee  hasA  Name
    :  :  :
    Staff  hasA  Name
    :  :  :
    Prof  hasA  Name
    Prof  hasA  EmpID
    :  :  :
```
What about

\[
\text{SemNet* (First Rel Second)}
\]

\[
\begin{array}{ccc}
: & : & : \\
\text{Sue} & \text{hasA} & \text{Name} \\
: & : & : \\
\text{Tomasz} & \text{in} & \text{Employees} \\
: & : & : \\
\text{Tomasz} & \text{teaches} & \text{Course} \\
: & : & : \\
\text{Prof} & \text{teaches} & 617 \\
: & : & : \\
\text{Employee} & \text{teaches} & \text{Course} \quad \text{no!} \\
: & : & : \\
\text{Tomasz} & \text{teaches} & 617 \quad \text{no!} \\
: & : & :
\end{array}
\]

We also don’t know

\[
\text{Tomasz teaches Sue in 617:}
\]

need a separate (ternary) relation.

In fact, \textit{SemNet} and \textit{SemNet*} could be seen as \textit{schemas}: relations containing other relation names, as \textit{metadata}.

T. H. Merrett
Syllogisms

\textbf{A, E} Universally quantified

\textbf{A} every X is Y \hspace{1cm} X \subseteq Y \hspace{1cm} Y' \subseteq X'

\textbf{E} no X is Y \hspace{1cm} Y \subseteq X' \hspace{1cm} X \subseteq Y'

\textbf{I, O} Existentially quantified

\textbf{I} some X is Y \hspace{1cm} X \emptyset Y \hspace{1cm} Y \emptyset X

\textbf{O} some X is not Y \hspace{1cm} X \emptyset Y' \hspace{1cm} Y' \emptyset X

- \textbf{A, E} are just the \textit{isA} hierarchy rules:
  - antisymmetric, transitive: \textit{.:} closure

- \textbf{I, O} combine with \textbf{A, E}: call \emptyset laps rules:
  - symmetric
  - \textit{X laps Y} & \textit{Y isA Z} \Rightarrow \textit{X laps Z}
    \hspace{1cm} (\textit{l laps} is closed under \textit{icomp} with \textit{isA})
  - \textit{X isA Y} & \textit{X isA Z} \Rightarrow \textit{Y laps Z}
  - \textit{X isA Y} \Rightarrow \textit{X laps Y}
Some syllogisms

<table>
<thead>
<tr>
<th>Barbara</th>
<th>Darii</th>
<th>Darapti</th>
<th>Disamis</th>
<th>Barbari</th>
</tr>
</thead>
<tbody>
<tr>
<td>( isA \ YZ )</td>
<td>( isA \ YZ )</td>
<td>( isA \ YZ )</td>
<td>( laps \ YZ )</td>
<td>( isA \ YZ )</td>
</tr>
<tr>
<td>( isA \ XY )</td>
<td>( laps \ XY )</td>
<td>( isA \ YX )</td>
<td>( isA \ YX )</td>
<td>( isA \ XY )</td>
</tr>
<tr>
<td>( isA \ XZ )</td>
<td>( laps \ XZ )</td>
<td>( laps \ XZ )</td>
<td>( laps \ XZ )</td>
<td>( laps \ XZ )</td>
</tr>
<tr>
<td>Celarent</td>
<td>Ferio</td>
<td>Felapton</td>
<td>Datisti</td>
<td>Celaront</td>
</tr>
<tr>
<td>( isA \ ZY' )</td>
<td>( isA \ ZY' )</td>
<td>( isA \ ZY' )</td>
<td>( isA \ YZ )</td>
<td>( isA \ ZY' )</td>
</tr>
<tr>
<td>( isA \ XY )</td>
<td>( laps \ XY )</td>
<td>( isA \ YX )</td>
<td>( laps \ YX )</td>
<td>( isA \ XY )</td>
</tr>
<tr>
<td>( isA \ ZX' )</td>
<td>( laps \ XZ' )</td>
<td>( laps \ XZ' )</td>
<td>( laps \ XZ )</td>
<td>( laps \ XZ' )</td>
</tr>
<tr>
<td>Cesare</td>
<td>Festino</td>
<td></td>
<td>Bocardo</td>
<td>etc.</td>
</tr>
<tr>
<td>( isA \ YZ' )</td>
<td>( isA \ YZ' )</td>
<td></td>
<td>( laps \ YZ' )</td>
<td></td>
</tr>
<tr>
<td>( isA \ XY )</td>
<td>( laps \ XY )</td>
<td></td>
<td>( isA \ YX )</td>
<td></td>
</tr>
<tr>
<td>( isA \ ZX' )</td>
<td>( laps \ XZ' )</td>
<td></td>
<td>( laps \ XZ' )</td>
<td></td>
</tr>
<tr>
<td>Camestres</td>
<td>Baroco</td>
<td></td>
<td>Ferison</td>
<td></td>
</tr>
<tr>
<td>( isA \ ZY )</td>
<td>( isA \ ZY )</td>
<td></td>
<td>( isA \ ZY' )</td>
<td></td>
</tr>
<tr>
<td>( isA \ YX' )</td>
<td>( laps \ YX' )</td>
<td></td>
<td>( laps \ YZ )</td>
<td></td>
</tr>
<tr>
<td>( isA \ ZX' )</td>
<td>( laps \ XZ' )</td>
<td></td>
<td>( laps \ XZ' )</td>
<td></td>
</tr>
</tbody>
</table>
Special (isA) syllogisms: logical closure

<table>
<thead>
<tr>
<th>S</th>
<th>Obj</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>K</td>
<td></td>
</tr>
</tbody>
</table>

1. No shark ever doubts that it is well fitted out.
2. A fish, that cannot dance a minuet, is contemptible.
3. No fish is quite certain that it is well fitted out, unless it has three rows of teeth.
4. All fishes, except sharks, are kind to children.
5. No heavy fish can dance a minuet.
6. A fish with three rows of teeth is not to be despised.

:. No heavy fish is unkind to children.

(Lewis Carroll, 1896, Symbolic Logic)
**Horn Clauses**  An Inference Engine

<table>
<thead>
<tr>
<th>[New]Facts (Concl)</th>
<th>Horn (Rule#)</th>
<th>Ante</th>
<th>Concl</th>
</tr>
</thead>
<tbody>
<tr>
<td>lays eggs</td>
<td>1</td>
<td>lays eggs</td>
<td>is bird</td>
</tr>
<tr>
<td>has feathers</td>
<td>1</td>
<td>has feathers</td>
<td>is bird</td>
</tr>
<tr>
<td>swims</td>
<td>2</td>
<td>flies</td>
<td>is bird</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>is not mammal</td>
<td>is bird</td>
</tr>
<tr>
<td>is bird</td>
<td>3</td>
<td>is bird</td>
<td>is duck</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>swims</td>
<td>is duck</td>
</tr>
<tr>
<td>is duck</td>
<td>3</td>
<td>is brown</td>
<td>is duck</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>is bird</td>
<td>is duck</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>swims</td>
<td>is duck</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>is green</td>
<td>is duck</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>is red</td>
<td>is duck</td>
</tr>
<tr>
<td>is duck</td>
<td>5</td>
<td>is duck</td>
<td>migrates</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>is not tame</td>
<td>migrates</td>
</tr>
</tbody>
</table>

*NewFacts is Facts ujoin

\[ \text{Concl} \in (\text{NewFacts}[\text{Concl} \supset \text{Ante}] \text{Horn}) \]*

*Relixpert* expands this 1-line inference engine to 50, in a 200-line expert system shell: *TDKE 6* (1991) 151

T. H. Merrett