Chapter 11: Knowledge Representation and Reasoning

- Many problems arose in earlier chapters were not resolved because they required knowledge of context.
- Two important aspects of context are general knowledge and specific knowledge.
- To analyze these we need the formalism for representing knowledge and reasoning. This area of study is called knowledge representation (KR)

Knowledge Representation and Reasoning

11.1 Knowledge Representation

There are two forms of knowledge that are crucial in any knowledge system:

✔general knowledge and specific knowledge.

✔*General knowledge:* type of hierarchy, part/whole relationships, and so on. For instance, it might code that **OWN1** is a relation between people and object but not individual person .

✔General world knowledge is essential for solving many language interpretation problems, one of most important being disambiguation.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

For example, the proper attachment of the final PP in the following two sentences dependents solely on the reader's background knowledge of the appropriate time needed for reading and for evolution:

I read a story about evolution in ten minutes
I read a story about evolution in the last million years

Specific Knowledge: is important for many issues, including determining the referent of noun phrases and disambiguating word senses based on what makes sense in the current situation.

- The knowledge representation is coding the knowledge and beliefs of the understanding system.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

- A knowledge representation consists of a database of sentences called the knowledge base (KB) and set of inference techniques , that can be used to derive new sentences given the current KB.
- The language in which the sentences in the KB are defined is called knowledge representation language (KRL). The KRL could be the same as the logical form, but there are practical reasons why they are often differ.
- In logical form language a wide range of quantifiers that is closely corresponding to the different word senses of English quantifiers.
- In the most current KRL, however there are usually only a few quantifiers and often only one construct allowing universal quantification.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

The KB must support two forms of inference: entail and implications.

Entail: formula P must be true given the formulas in KB or given the formulas representing the meaning of a sentence, then we say that the KB (or sentence) entails P.

Implications are conclusions that can typically be drawn from a sentence but that could be explicitly denied in specific circumstances.

Example: ***Jack owns two cars -*** entails that:

Jack owns a car (that is, this fat cannot be denied).

But only implies that he does not own three cars, as we could continue by saying ***In fact, he owns three cars.***

Knowledge Representation and Reasoning

11.1 Knowledge Representation

Types of Inference:

Many different forms of inference are necessary to understand natural language.

Inference techniques can be classified into deductive and non-deductive forms.

Deductive inference: give a set of facts, a deductive inference process will make only conclusions that logically follow from those facts.

Non-deductive inference falls into several classes: *inductive inference* and *abductive reference.*

Inductive inference is learning generalities from examples.

Abductive reference is inferring causes from effects.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

Abductive inference can be contrasted with deductive inference by considering the axiom:

$$\mathbf{A \supset B}$$

deductive inference would use this axiom to infer B when given A.

Abductive inference would use it to infer A given B, since A is reason that B is true.

Many systems allow the use of default information. The default rule is an inference rule to which there may be exceptions, thus it is defeasible.

If we write a default information using notation **A ⇒ B,** then the default inference rule could be stated as follows: if A ⇒ B, and a is true, and] B is not provable, then conclude B.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

Example: Given sentence ***Birds fly***, could be represented by the FOPC formula:

$$\forall x \; \text{BIRD}(x) \Rightarrow \text{FLIES}(x)$$

This has the effect that whenever there is a bird B for which is not provable that $\rceil$ FLIES (B), then it can be inferred that FLIES (x).

Defeasible rule introduce a new set of complexities in a representation without such rules, most representations are monotonic, because adding a new assertion only increases the number of formulas entailed.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

For instance, if the knowledge base KB1 entails a conclusion C, and if we add an additional formula to KB1 to form a new consistent knowledge base KB2, the KB2 will also entail C.

This is not true of representation that uses defeasible rules, and hence they are called nonmonotonic representation.

Consider a knowledge base KB consisting of the formulas:

Cat(Sampson)	□ Sampson is a cat
TabbyCat(Sampson)	□ Sampson is a tabby cat
$\forall$ c. Cat(c) $\Rightarrow$ Purrs(c)	Cats purr

Given this KB we can conclude Purrs (Sampson) using the default rule because there is no information to contradict Purrs(S).

Knowledge Representation and Reasoning

11.1 Knowledge Representation

On the other hand, if we add a new fact that no tabby cats purr, then the extended knowledge have would no longer entail that Sampson purrs.

Inference Techniques

- The two main classes of inference techniques found in knowledge representation systems are procedural and declarative.

- Most systems combine these techniques to some extent, forming a continuum from purely declarative representation to purely procedural ones. .

Knowledge Representation and Reasoning

11.1 Knowledge Representation

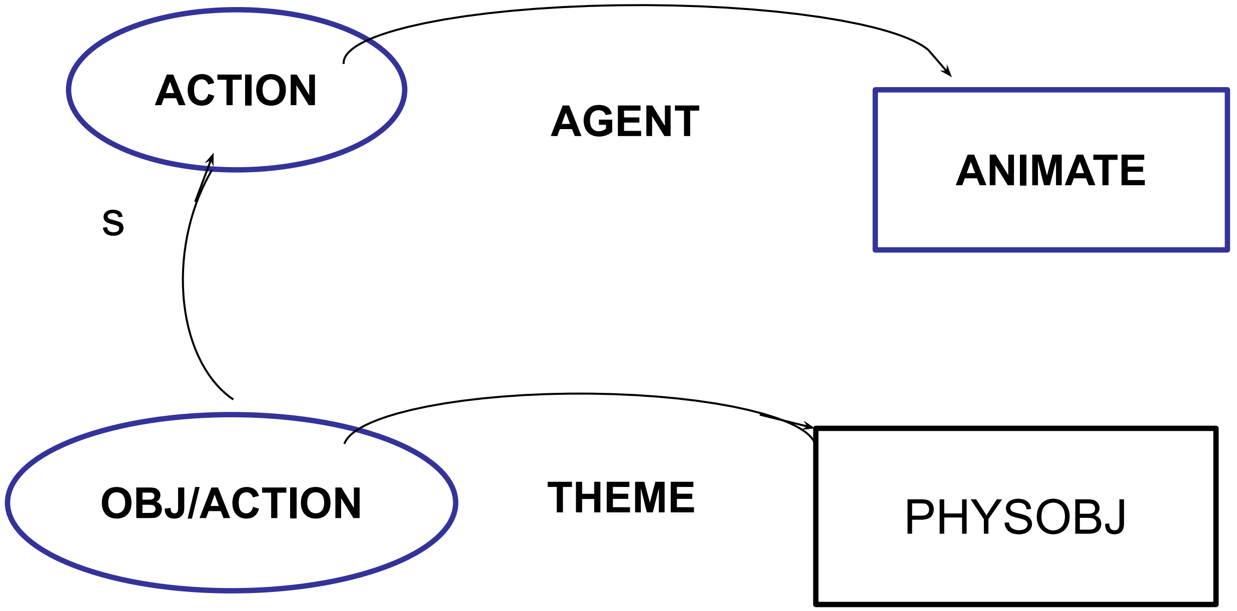

Figure 11.1 An example of simple inheritance

Knowledge Representation and Reasoning

11.1 Knowledge Representation

In chapter 10 the techniques of inheritance was introduced for semantic networks. This inference process can be realized procedurally or declaratively.

Given simple network in figure 11.1, the following FOPC axioms might represent this information.

1. $\forall x . \mathrm{ACTION}(x) \supset \exists a . \mathrm{AGENT}(x, a) \;\&\; \mathrm{ANIMATE}(a)$
2. $\forall a \exists x . \mathrm{ACTION}(x) \;\&\; \mathrm{AGENT}(x, a) \supset \mathrm{ANIMATE}(a)$
3. $\forall x . \mathrm{OBJ/ACTION}(x) \supset \mathrm{ACTION}(x)$
4. $\forall x . \mathrm{OBJ/ACTION}(x) \supset \exists o . \mathrm{THEME}(x, o) \;\&\; \mathrm{PHYSOBJ}(o)$
5. $\exists o . \exists x . \mathrm{OBJ/ACTION}(x) \;\&\; \mathrm{THEME}(x, o) \supset \mathrm{PHYSOBJ}(o)$

Knowledge Representation and Reasoning

11.1 Knowledge Representation

Declarative system

Using these axioms, we can prove that the class OBJ/ACTION *inherits* the AGENT role. In other words, for any A such that OBJ/ACTION(A) is true, we could prove that A has an AGENT role, that is (by using axioms 3 and 1):

$$\exists\ a\ .\ AGENT(A\ ,\ a)\ \&\ ANIMATE(a)$$

Procedural system (1)

As described in chapter 9, a procedural version of this would be a program that start at the specified node OBJ/ACTION, finds all roles attached at that node and the follow the S arc up to the supertype ACTION and finds all the roles attached there.

Knowledge Representation and Reasoning

11.1 Knowledge Representation

Procedural system (2)

The complete set of the roles gathered by this procedure is the answer. Thus any OBJ/ACTION has an AGENT role inherited from the class ACTION.

Note that:

- Both these techniques compute the same result, but the first does it by deduction over logical formulas, while the second uses a program that performs a graph traversal.
- The first technique seems more rigorously defined, but the second is probably more efficient.

11.2 A Representation Based on FOPC

- KRL will be an extended version of the first-order predicate calculus.
- Note that by choosing the language , we are not committed to any particular form of inference.
- We will focus on the extensions to standard FOPC that are needed to represent the meaning of the natural language sentences.

+ The terms in the language are constants (such as *John1*), functions (such as *father(John1)*), and variables (such as *x* and *y*).

+ Logical form language did not use constants.

Knowledge Representation and Reasoning

11.2 A Representation Based on FOPC

Example: - logical form term (NAME j1 "John")

- in KB, actual person referred to in a given context might be represented by the constant *John1*.

- Restricted quantification are used in the KRL that are similar to the generalized quantifier notation in the logical form language.
- Restrictions follow the quantifier variable separated by ":".

Example: $\exists$ x : Man (x) Happy (x) $\longleftrightarrow$ $\exists$ x . Man (x) & Happy (x)

$\forall$ x : Man (x) Happy (x) $\longleftrightarrow$ $\forall$ x . Man (x) $\supset$ Happy (x)

Knowledge Representation and Reasoning

11.2 A Representation Based on FOPC

- The *equality* predicate (a = b), which states that terms a and b have the same denotation.
- Given simple proposition Pa involving a constant a, if Pa is true and a = b, then Pb must be true as well, where Pa is the same as Pb, except that a has been replaced b

Knowledge Representation and Reasoning

11.3 Frames: Representation Stereotypical Information

(Study oneself)

Knowledge Representation and Reasoning

11.4 Handling Natural Language Quantifications

- We can now consider some issues in mapping the logical form language into the KRL.
- One of the most obvious differences between the two languages is the treatment of quantifiers.
- Significant progress can be made to reduce the differences, however by extending the ontology of the KRL to allow sets as objects.
  + A set is a collection of objects as a unit.
  + Sets may be finite (such as the set of *John* and *Mary*) or

infinite (such as the set of numbers greater than 7). We will only use a finite sets in KRL.

  + A set can be indicated by listing its members in curly brackets

Knowledge Representation and Reasoning

11.4 Handling Natural Language Quantifications

Example: *{ John1, Mary1}*. That order does not matter;

{John1, Mary1} = {Mary1, John1}.

- We also allow constants to denote sets. Thus S1 might be a set defined by the formula S1= *{ John1, Mary1}*.

- Sets might be members of other sets.
- Sets will usually be defined in terms of some property. This will be written in the form $\{y \mid Py\}$.
- In addition, the following predicates to relate sets and individuals:
- the following predicates to relate sets and individuals;

$S1 \subset S2$	iff all the elements of S1 are in S2
$x \in S$	iff x is a member of the set S

Knowledge Representation and Reasoning

11.4 Handling Natural Language Quantifications

An interpretation for *Some men met at three,* as follows:

$$\exists\ M : M \subset \{ x \mid Man\ (x)\}\ .\ Meet1\ (M, 3\ PM).$$

- Consider the different formulas that arise from the collective/distributive readings.

Example: There are two interpretations of the sentence *Some men bought a suit,* which has following logical forms (omitting the tense operator)

```
( SOME  m1 : ( PLUR  MAN1 )
 ( A  s1 : SUIT1
 BUY1  m1  s1 )))
```

Knowledge Representation and Reasoning

11.4 Handling Natural Language Quantifications

- The collective reading would map to:

  $\exists\ M1 : M1 \subset \{ z \mid Man(z) \}\ \ \exists\ s : Suit(s)\ .\ Buy1(M1, s)$

  that is, there is a subset of the set of all men who together bought a suit.

- The distributive reading would map to:

  $\exists\ M2 : M2 \subset \{ z \mid Man(z) \}\ \ \forall\ m : m \in M2$
  $\quad \exists\ s : Suit(s)\ .\ Buy1(m, s)$

  that is, there are some men individually buying suit .

Note that the collective and distributive readings involve a common core meaning involving the subset of men The only difference is whether we use the set as a unit or quantify over all members of the set.

Knowledge Representation and Reasoning

11.4 Handling Natural Language Quantifications

New function of cardinality of set

For any given set S, let | S | be the number of elements in S.

For example, the meaning of *Three men entered the room* would be as follows (tense information is omitted):

$$\exists\ M : (M \subset \{ y \mid Man (y) \} \ \& \ | M | = 3)$$
$$\forall\ m : m \in M \,.\, Enter1 (m , Room1)$$

By changing the restriction to | M | ≥ 3, we get the meaning of *At least three men entered the room*.

If we define most as being true if more than half of some set has a given property, then *Most men laughed* might have the meaning:

Knowledge Representation and Reasoning

11.4 Handling Natural Language Quantifications

If we define most as being true if more than half of some set has a given property, then *Most men laughed* might have the meaning:

$$\exists M: (M \subset \{y \mid Man(y)\} \& |M| \geq |\{y \mid Man(y)\}|$$

$$\forall m : m \in M . Laughed(m)$$

Knowledge Representation and Reasoning

11.5 Time and Aspectual Classes of Verbs

- One of the central components of any knowledge representation that supports natural language is the treatment of verbs and time.

Time

- In the logical form language, temporal information was handled in

several ways. There were modal operators to present tense: PAST, PRES, PROG, FUT, and temporal connectives: PEFORE, DURING.

- There are several different types of time: time point, interval, durations.

- The following predicates are allowed for temporal relations:

t1 < t2 point / interval t1 is before point / interval t2

Knowledge Representation and Reasoning

11.5 Time and Aspectual Classes of Verbs

t1 : t2   interval t1 meets interval t2, or point t1 defines the beginning of the interval t2, or point t2 defines the end of interval t1.

$t1 \subseteq t2$ point/ interval t1 is contained in interval t1

Some predicates can be true only over interval of times, whereas others can be true only at points, and other can be true at either.

Aspectual Classes of Verbs

- Sentences describe propositions that fall into at least three distinct classes:

  + those that describe states (stative proposition);

  + those define ongoing activities (activity proposition);

  + those define completed events (telic proposition)

Knowledge Representation and Reasoning

11.5 Time and Aspectual Classes of Verbs

Example:

- Sentences describe states: *Jack is happy.*

  I believe the world is flat.

- Sentences describe activities:: Jack is running.

  The door was swinging to and fro.

- Sentences define completed events: *Jack fell asleep.*

  They climbed the mountain in two days.

Encoding Tense

Tense operators can also be represented directly in the temporal logic without the need for modal operators.

Knowledge Representation and Reasoning

11.5 Time and Aspectual Classes of Verbs

- Let's assume that the constant NOW denotes the current time.
- Given this we could map the PAST operator into a formula that existentially quantifies over a time before now.
- The sentence *John was happy* would map to the KR expression:

$$\exists\ T1 . T1 < NOW1 . Happy (Jack1, T1)$$

- *The same sentence in the simple present* *Jack is happy*:

$$\exists\ T1 . NOW1 \subseteq T1 . Happy (John1 , NOW1)$$

- And the simple future *Jack will be happy:*

$$\exists\ T1 . T1 > NOW1 . Happy (Jack1, T1)$$

Knowledge Representation and Reasoning

11.5 Time and Aspectual Classes of Verbs

Specifically, Reichenbach (1947) developed a theory that tense gives information about three times:

S –the time of speech

E – the time of event /state

R - the reference time

Example:	*Jack sings*	simple present :	$S = R$, $E = R$
	Jack sang	simple past :	$R < S$, $E = R$
	Jack will sing	simple future :	$S < R$, $E = R$

(see at figure

Knowledge Representation and Reasoning

11.5 Time and Aspectual Classes of Verbs

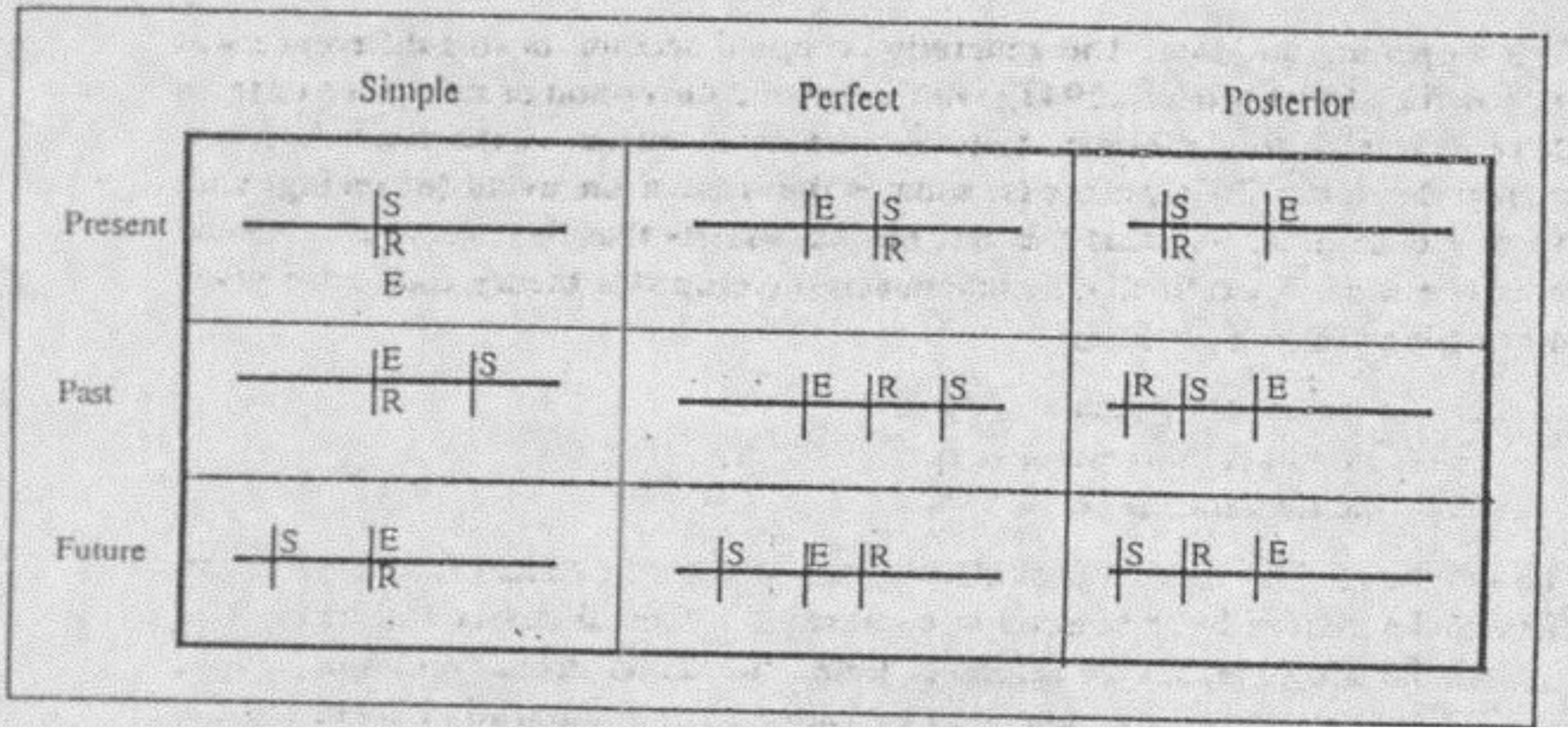

Figure 11.2: Some temporal configurations allowed by the tenses

Knowledge Representation and Reasoning

11.6 Procedural Semantics and Question Answering

- Procedurally based techniques are frequently used in database query applications,
- where there is a large difference in expressive power between the logical form language and database language.
- Rather than convert the logical form language into extended FOPC , the logical forms are treated as expressions in a query language.

Example: the query *Does every flight to Chicago serve breakfast ?* *with logical form:*

(EVERY f1 : (& (FLIGHT f1) (DEST f1 (NAME c1 “Chicago”)))

(SERVE – BREAKFAST f1))

would be interpreted as a procedure as follows:

Knowledge Representation and Reasoning

11.6 Procedural Semantics and Question Answering

1. Find all flights in the database with destination CHI (the database symbol for Chicago);
2. For each flight found, check if it serve breakfast. If all do, return yes, otherwise return no.

- To interpret logical form expression as procedure, a method of interpretation often called procedural semantics.

- Two database functions are assumed:

(Test<literal>1 … , <literal> n) - return true if there is some binding of the variables such that each literal is found in the database.

Knowledge Representation and Reasoning

11.6 Procedural Semantics and Question Answering

(Retrieve < var> <literal>1 ... , <literal>n) – like Test, but if it succeeds it returns every instance of the indicated variable that provides a solution.

- For example, given the database in figure 11.3, the query:

(Retrieve ? x (FLIGHT ? x) (ATIME ? x CH1 1000 HR))

will return the list (F2) because F2 is only binding of ?x where both these literals are in the database.

- The procedural semantics approach gets more interesting as it interprets logical connective and quantifiers, which have no corresponding construct in the relational database.

```
(FLIGHT F1)          (ATIME F1  CHI  1700HR)
(FLIGHT F2)          (ATIME F2  CHI  1000HR)
(FLIGHT F3)          (ATIME F3  CHI   900HR)
(FLIGHT F4)          (ATIME F4 BOS  1700HR)
(AIRPORT BOS)        (DTIME F1  BOS  1600HR)
(AIRPORT CH)         (DTIME F2  BOS   900HR)
                     (DTIME F3  BOS   800HR)
                     (DTIME F4  CHI   1600HR
```

H.11.3 A simple database of airline schedules

Knowledge Representation and Reasoning

11.6 Procedural Semantics and Question Answering

- The logical operators are interpreted as follows.

  + Conjunctions (& R1 , ... Rn) – will translate into a program of the form (CHECK – ALL – TRUE (R1) , ... T (Rn))

  + Disjunctions: (OR R1 , ... Rn) – will translate into a program of the form (FIND – ONE – TRUE T (R1) , ... , T (Rn))

- The procedure for negation is interpreted as follows.

  (NOT R) – translates into a program of the form (UNLESS T(R)).

Knowledge Representation and Reasoning

11.6 Procedural Semantics and Question Answering

- Three quantifiers important in question-answering applications: THE, EACH, and WH

  + (THE x : Rx Px) – translates into a program (FIND –THE ?x T(R ?x) T (p ?x))

  + (EACH x : Rx Px) – translates to a program (ITERATE ?x T (R ? x) T (P ? x))

- + (WH x : Rx Px) – translates to a program PRINT – ALL ?x

T(R ? x) T (P ? x))

To show some examples using the database in figure 12.3.

Knowledge Representation and Reasoning

11.6 Procedural Semantics and Question Answering

Example: the query *Which flight to Chicago leaves at 4 PM ?* would have the logical form (after scoping)

(WH f1 : (& (FLIGHT f1) (DEST f1 (NAME c1 “Chicago”))) (LEAVE l1 (NAME t1 “4 PM”)))

This would translate into a query of the form

(PRINT – ALL ? f (FLIGHT ? f) (ATIME ? f CH1 ? t) (DTIME ? f ? s 1600 HR))

The DEST relation maps to an ATIME relation and the LEAVE predicate maps into the DTIME relation. Note that the departure location was not specified in the logical form and so is treated as a variable here.

EXERCISE OF CHAPTER 11

1) Give a plausible logic-based representation for the meaning of the following sentences. Focusing on the interpretation of the quantifiers. If the sentence has a collective/distributive ambiguity, give both interpretations

   Several men cried

   Seven men in the book met in the park.

   All but three men bought a suit.

2) Given the sentence *What arrived time of each flight from Boston is ?.*

a) Translate the query into a logical form.

b) Translate the logical form of a) into a procedural semantics based on the database in figure 11.3.

c) Present step by step how the computer performs the procedural semantics to export the answer.