An Example of an Interpretation for Definite Clauses

CSC384 Supplemental Handout
Courtesy Craig Boutilier

January 13, 2003

Suppose we wish to express our knowledge about the courses being taught by people in the CS department, perhaps for the purposes of scheduling. We might have a language with the following symbols (and perhaps more):

1. Constant symbols: `craig`, `fahiem`, `kyros`, `cs148`, `cs238`, `cs324`, `cs384`, `pratt`, `sidsm`, `bahen`, etc.

2. Function symbols: `instructor` (with arity 1), `location` (with arity 1)

3. Predicate symbols:

   (a) Predicates with arity 0 (atomic propositions): `sunny`, `rainy`
   (b) Predicates with arity 1 (properties): `course`, `teacher`, `inshape`, `busy`, `soggy`, `tanned`
   (c) Predicates with arity 2 (relations): `teaches`, `distinct`

While the symbols above have no inherent meaning (that is what an interpretation is for), we clearly have an intended meaning. The constant symbols refer to the individuals (teachers, courses and buildings) in our domain. The function `instructor`, when applied to a course, returns the instructor of that course. The function `location`, when applied to a course, says where that course is held, and when applied to an instructor, says where the instructor's office is (in either case, the location should be a building). The predicates `course` and `teacher` are true of individuals that are courses or teachers, respectively. The predicates `inshape`, `busy`, `soggy`, `tanned` are intended to refer to properties of instructors. The predicate `teaches` is a relation between teachers and courses, while `distinct` is a predicate that tells us whether two individuals (either courses or buildings) are different.

We might write, for instance, the following clauses:

```
busy(craig) <- teaches(craig,cs384) & teaches(craig,cs148).
busy(craig) <- teaches(craig,cs384) & teaches(craig,cs324).

(or more general versions of this...)

busy(craig) <- teaches(craig,X) & teaches(craig,Y) & distinct(X,Y).

busy(Z) <- teaches(Z,X) & teaches(Z,Y) & distinct(X,Y).

---

inshape(X) <- teaches(X,Y)  & distinct(location(X),location(Y)).

soggy(X) <- teaches(X,Y)  & distinct(location(X),location(Y)) & rainy.

tanned(X) <- teaches(X,Y)  & distinct(location(X),location(Y)) & sunny.

(and so on...)
```

We might also have clauses that are *facts* like: `course(cs384)`, `course(cs148)`, `teaches(craig,cs384)`, `distinct(location(craig),location(cs148))`, `sunny`, etc.

Let's consider one possible interpretation for this language.

First we need a domain $\mathcal{D}$ or set of individuals: for this we will use the following set of elements:

$$\mathcal{D} = \{craig, fahiem, kyros, cs148, cs238, cs324, cs384, pratt, sidsm, bahen\}$$

Note: these are members of our domain, *not* constant symbols in our language.

Now, we need a mapping ϕ that determines the denotation of our constant symbols and function symbols. Recall, one way to think about ϕ is as a mapping, for all n, $\phi : \mathcal{F}^n \times \mathcal{D}^n \to \mathcal{D}$. (Remember, $\mathcal{F}^n$ is the set of function symbols with arity n and we think of $\mathcal{F}^0$ as the set $\mathcal{C}$ of constant symbols.) One possible mapping might be as follows:

1. The constant symbols will be mapped into the obvious members of $\mathcal{D}$; e.g.,

   - $\phi(\texttt{craig}, \langle\rangle) = craig$
   - $\phi(\texttt{cs384}, \langle\rangle) = cs384$
   - etc.

   Note, that since constants take no arguments, we have only one 0-tuple as the second argument to ϕ.

2. The function symbol `instructor` will be interpreted as follows:

   - $\phi(\texttt{instructor}, \langle cs148 \rangle) = craig$
   - $\phi(\texttt{instructor}, \langle cs238 \rangle) = fahiem$
   - $\phi(\texttt{instructor}, \langle cs324 \rangle) = kyros$
   - $\phi(\texttt{instructor}, \langle cs384 \rangle) = craig$

   Note: we have not specified how `instructor` is interpreted when applied to domain individuals like buildings. Technically, an interpretation must say what $\phi(\texttt{instructor}, \langle bahen \rangle)$ is; but we will never talk about this, so we will won't bother to specify what this interpretation has to say about it (feel free to complete the description as you see fit). Also note: $\langle cs148 \rangle$ is a collection of (one) domain elements, it is *not* a constant symbol in the language.

3. The function symbol `location` will be interpreted as follows:

   - $\phi(\texttt{location}, \langle cs148 \rangle) = pratt$
   - $\phi(\texttt{location}, \langle cs384 \rangle) = sidsm$
   - $\phi(\texttt{location}, \langle craig \rangle) = sidsm$
   - $\phi(\texttt{location}, \langle cs238 \rangle) = bahen$
   - etc.

   Feel free to fill in more details of the interpretation.

Now, we need a mapping π that determines the denotation of our predicate symbols. Recall, one way to think about π is as a mapping, for all n, $\pi : \mathcal{P}^n \times \mathcal{D}^n \to \{\top, \bot\}$. (Remember, $\mathcal{P}^n$ is the set of predicate symbols with arity n.) One possible mapping might be as follows:

1. The predicate symbols `rainy` and `sunny` will be interpreted as follows:

   - $\pi(\texttt{rainy}, \langle\rangle) = \top$
   - $\pi(\texttt{sunny}, \langle\rangle) = \bot$

   Note, that since these symbols take no arguments, we have only one 0-tuple as the second argument to π.

2. The predicate symbol `course` will be interpreted as follows:

- $\pi(\text{course}, \langle cs148 \rangle) = \top$
- $\pi(\text{course}, \langle cs238 \rangle) = \top$
- $\pi(\text{course}, \langle cs324 \rangle) = \top$
- $\pi(\text{course}, \langle cs384 \rangle) = \top$
- $\pi(\text{course}, \langle craig \rangle) = \bot$
- $\pi(\text{course}, \langle sidsm \rangle) = \bot$
- etc.

We might write this more simply by saying that the *extension* of the predicate is $\{cs148, cs238, cs324, cs384\}$. This is the set of domain elements for which the property denoted by the symbol `course` is true.

3. The predicate symbol `teaches` will be interpreted as follows:

- $\pi(\text{teaches}, \langle craig, cs148 \rangle) = \top$
- $\pi(\text{teaches}, \langle craig, cs238 \rangle) = \bot$
- $\pi(\text{teaches}, \langle craig, cs324 \rangle) = \bot$
- $\pi(\text{teaches}, \langle craig, cs384 \rangle) = \top$
- $\pi(\text{teaches}, \langle kyros, cs148 \rangle) = \bot$
- etc.

Once again, we can more simply say that the extension of the predicate denoted by the symbol `teaches` is

$$\{\langle craig, cs148 \rangle, \langle craig, cs384 \rangle, \langle fahiem, cs238 \rangle, \langle kyros, cs324 \rangle\}$$

4. The predicate symbol `inshape` denotes a predicate with the extension $\{craig, fahiem\}$; while `soggy` has extension $\{craig\}$.

5. The predicate symbol `distinct` has the following elements in its extension: $\langle cs148, cs238 \rangle$, $\langle cs148, cs324 \rangle$, $\langle cs148, cs384 \rangle$, $\langle cs238, cs324 \rangle$, etc. together with $\langle sidsm, pratt \rangle$, $\langle sidsm, bahen \rangle$, $\langle pratt, sidsm \rangle$, $\langle pratt, bahen \rangle$, $\langle bahen, pratt \rangle$ and $\langle bahen, sidsm \rangle$. This ensures that each building is distinct from each other building, and that each course is distinct from each other course.

Please finish by supplying the details of how this interpretation might interpret the rest of the symbols. Consider the clauses above and ask if this interpretation is a model for those clauses.