A Representation and Reasoning System (RRS) is made up of:

- **Formal language (syntax):** Specifies the legal sentences (the range of things that can be said).
- **Semantics:** Specifies the meaning of the symbols (for your domain) and specifies the rules of inference (i.e., the range of things that can be said).
- **Reasoning theory or proof procedure:** Specifies how an answer can be produced. Can be deterministic or non-deterministic.

Implementation of an RRS:

- **Reasoning procedure:** Resolves the nondeterminism of the reasoning theory.

Implementation of an RRS:

- **Reasoning procedure:** Resolves the nondeterminism of the reasoning theory.

**Previous Class**

- Introduced task domains: robot delivery and wiring.
- Introduced the symbolic approach.

- Symbols have meaning to the knowledge engineer.
- Symbols are used in the knowledge engineer.
- Symbols are used to build a knowledge base that the computer is told about.
- The computer reasons with the facts and rules to make new conclusions.

**Overview**

- Two views of semantics:
  - Logical consequence
  - Models
  - Adding variables to semantics
  - Semantics of Datalog

Representation and Reasoning System
Overview

• Representation and Reasoning System
  ⇒ Syntax of Datalog

• Semantics of Datalog
  • Adding Variables to Semantics
  ⇒ Models

• Models
  • Syntax of Datalog

• Logical Consequence

Simplifying Assumptions of Initial RRS

• Each individual can be given a unique name
• Only a finite number of individuals of interest in the domain
• The environment is static
• An agent's knowledge base consists of definite and positive
  individuals and relations among individuals
• An agent's knowledge can be usefully described in terms of

Different RRS's

• Different RRS's good for different domains
• Different RRS's good for different domains
• Different RRS's good for different domains

Choose the simplest RRS possible for your application.
Example

- What are the constants?
- What are the predicate symbols?
- What are the variables?
- Whether knowledge base is correct depends on semantics

Knowledge Base

- More Syntax of Datalog

- Variable - starts with upper-case letter
- Constant - starts with lower-case letter or is a sequence of digits (numeric)
- Predicate symbol
- Atomic symbol (atom) - of the form \( p(t_1, \ldots, t_n) \) where \( p \) is a predicate symbol and \( t_i \) are terms
- Term - either a variable or a constant
- Const
- Suffix with lower-case letter
- String with upper-case letter
- Empty
Interpretation

An interpretation is a triple \( I = (D, \phi, \pi) \) where

- \( D \) the domain, is a nonempty set. Elements of \( D \) are individuals
- \( \phi \) maps each constant to an element of \( D \)
- \( \pi \) maps each predicate symbol to a subset of \( D^n \)

\( \phi(c) \) denotes individual in \( D \)

Semantics

Semantics concerns two things

- A domain, and a mapping from the syntax to the domain
- How constants and predicate symbols in the syntax correspond to the individuals and relations in the domain

Two Views of Semantics

Logical Consequence

Models

Adding variables to Semantics

Semantic of Datalog

Syntax of Datalog

Representation and Reasoning Sysem

Overview
Example Interpretation

- $D$ is the set of people: William, George, Sally.
- It is the actual people, not the names.
- Knowledge Engineer decides $D$ and mapping of all constants to $D$.
- Knowledge Engineer decides on mapping of all predicates.

Example Interpretation Cont'd

- Knowledge Engineer decides on mapping of all predicates.
- ${\text{male}}$ maps to $\{\text{<William>}, \text{<George>}\}$.
- ${\text{female}}$ maps to $\{\text{<Sally>}\}$.
- Thus, there is no mapping from the sex predicate $\text{male}$ to $\text{female}$.
- William and George are male. Sally is female.

Example continued

- How many different $\phi$?
- Domain with $\{x \land \forall x A(x)\}$.
- If $\phi$ includes with constants $a$ and $p$ and 1-ary predicate $\text{female}$(x)
- Example: focus on all interpretations, not just intended one.
- Example: William and George are male, Sally is female.
- How many different $\phi$'s?
- $\phi_1(a)$
- $\phi_2(b)$
- $\phi_3$...
- $\phi_9$...
Semantics of Connectives

\[ p \land q \]
\[ p \rightarrow q \]
\[
\begin{array}{ccc}
\text{true} & \text{true} & \text{true} \\
\text{true} & \text{false} & \text{false} \\
\text{false} & \text{true} & \text{false} \\
\text{false} & \text{false} & \text{false} \\
\end{array}
\]

Nota bene:
- \( p \rightarrow q \) is true when both \( p \) and \( q \) are false.
- \( p \land q \) doesn't always correspond to 'english' meaning.

Thus \( h \rightarrow b_1 \land \cdots \land b_m \) is false in interpretation \( I \) if \( h \) is false in \( I \) and each \( b_i \) is true in \( I \).

Semantics of Ground Atoms from Interpretation

\[ \{ \} \] is true if the set of elements is empty.
So it maps to true.

\[ \{ < \text{George} >, < \text{William} > \} \]
\[ \{ < \text{George} >, < \text{William} > \} = (\text{male} \circ \text{doctor} \circ \text{name}) (\text{George}) \]
\[ \{ < \text{George} >, < \text{William} > \} \]
\[ \text{William} \text{ does } \text{not } \text{like } \text{George} \]
\[ \text{George} \text{ does } \text{not } \text{like } \text{William} \]

\[ \phi \text{ maps to true if } \exists \text{ } \eta \in \pi \text{ such that } (\phi) \eta \]
\[ \text{otherwise maps to false} \]

Ground atom has no variables.

Determining Truth of Ground Atoms in \( I \)

\[ \{ \} \]
\[ \{ \} \]
\[ \{ \} \]
\[ \{ \} \]

\[ \{ \} \]

\[ \{ \} \]

\[ \{ \} \]

\[ \{ \} \]

\[ \{ \} \]

\[ \{ \} \]

How many different interpretations are there altogether?

\[ \begin{array}{c|c}
\phi & \text{there are at } \phi \\
\hline
\text{false} & \text{false} \text{ formulas} \\
\text{true} & \text{true} \\
\text{false} & \text{false} \\
\text{true} & \text{true} \\
\end{array}
\]

How many, \( \neq \)?

Example Continued
Overview

- Representation and Reasoning System
- Syntax of Datalog
- Semantics of Datalog
- Adding Variables to Semantics
- Models
- Logical Consequence
- Two Views of Semantics

Limitations of Datalog

- Datalog does not include an operator that means negation
- Cannot write a rule that ensures just one of male and female is true for any person
- Even if every object is male or female, both predicates needed
- Male(exists) ∧ Female(exists) → undefined

Example

- Is male(exists) ∧ Female(exists) true in I?
- Is male(exists) → Female(exists) true in I?
- Is male(exists) → Female(exists) true in I?
- Is male(exists) ∧ Female(exists) true in I?
Semantics & Variables

• How do we interpret clauses such as
  \( \text{person}(X) \leftarrow \text{female}(X) \)

• Clause is true if it is true for all values of \( X \)
  - \( \text{person}(X) \) must be true whenever \( \text{female}(X) \) is true
  - Remember, knowledge engineer had to specify mapping for all predicates, even room
  - \( \pi(\text{female}) \subseteq \pi(\text{person}) \)

• It really has a universal quantifier
  - For all \( X \) \( \text{female}(X) \leftarrow \text{person}(X) \)

• So, variables have an implicit universal quantifier over the clause

Variable Assignment: Formal Definition

• Define a variable assignment \( \rho \)
  - Maps each variable to some object in the domain

• Together \( \rho \) and \( \phi \) assign each term to some object in the domain

• Together \( \rho \) and interpretation \( I \) map every clause to true or false
  + Even ungrounded ones

• Now we can say:
  - A clause is true in an interpretation if it is true for all variable assignments

Example

• Interpretation I
  - \( \pi(\text{male}) = \{<\text{William}>, <\text{George}>\} \)
  - \( \pi(\text{female}) = \{<\text{Sally}>\} \)
  - \( \pi(\text{person}) = \{<\text{William}>, <\text{George}>, <\text{Sally}>\} \)

• Are the following true?
  \( \text{person}(X) \leftarrow \text{male}(X) \)
  \( \text{person}(X) \leftarrow \text{female}(X) \)
  \( \text{male}(X) \land \text{female}(X) \)
  \( \text{male}(X) \lor \text{female}(X) \)
  \( \text{person}(X) \leftarrow \text{female}(X) \land \text{male}(\text{william}) \)
Models

A model of a set of clauses is an interpretation in which all the clauses are true.

Example KB:

- KB: p ← q.
  q.

π[p] (π[q] (π[p ← q])

- Model of KB:
  I₁: TRUE TRUE
  I₂: TRUE FALSE
  I₃: FALSE TRUE
  I₄: FALSE FALSE

Sets of Clauses

A set of clauses is true in an interpretation if each clause is true in the interpretation.

- Note that we universally quantify for the variables over each clause.
- In other words, if two clauses use the same variables, it is the same as if they used different variables.
- In each clause, all the variables are included.
- In each clause is true in the interpretation.

A set of clauses is true in an interpretation.

Two Views of Semantics

- Logical Consequence
  - Models
  - Adding Variables to Semantics
  - Semantics of Datalog
  - Syntax of Datalog

Representation and Reasoning System
Logical Consequence

If $KB$ is a set of clauses and $g$ is a conjunction of atoms, $g$ is a logical consequence of $KB$, written $KB \models g$, if $g$ is true in every model of $KB$.

This tells us that our $KB$, by its definition, always forces $g$ to be true.

Other terms that mean the same thing:

- $g$ logically follows from $KB$
- $KB$ entails $g$

If $KB$ is a logical consequence of $KB$, written $KB \models g$. If $KB$ is not a logical consequence of $KB$, written $KB \not\models g$.

If $KB$ is a set of clauses and $g$ is a conjunction of atoms.

Example with constants

Example:

(focus on all interpretations, not just intended one)

Language with constants $a$ and $b$ and 1-ary predicate $girl(x)$

Domain with $D = \{x, y, z\}$

There are 9 $\phi_i$'s and 8 $\pi_j$'s, so 72 interpretations.

How many models of $KB = \{girl(a), girl(b)\}$?

Checking each would take too long, so let's break down into subcases

Case 1:

$\phi_i(a) = \phi_i(b)$

How many of the 9 $\phi_i$'s have $\phi_i(a) = \phi_i(b)$?

When $\phi_i(a) = \phi_i(b) = x$, which $\pi_j$'s make $KB$ true?

So how many models of $KB$ with $\phi_i(a) = \phi_i(b)$?

Case 2:

$\phi_i(a) \neq \phi_i(b)$

How many of the 9 $\phi_i$'s have $\phi_i(a) \neq \phi_i(b)$?

When $\phi_i(a) = x$ and $\phi_i(b) = y$, which $\pi_j$'s make $KB$ true?

So how many models of $KB$ with $\phi_i(a) \neq \phi_i(b)$?

Two Views of Semantics

- Logical Consequence
- Models
- Adding Variables to Semantics
- Semantics of Datalog
- Syntax of Datalog
- Representation and Reasoning System

Example with constants

How many models of $KB$?

Checking each would take too long, so let's break down into subcases

How many models of $KB$ with $\phi_i(a) \neq \phi_i(b)$?

Example: Focus on all interpretations, not just intended ones.

Example with constants

How many models of $KB$ with $\phi_i(a) \neq \phi_i(b)$?
Ask questions about your domain
- Choose a task domain: intended interpretation
- Associate constants with individuals you want to name
- For each relation you want to represent, associate a predicate symbol in the language
- Tell the system clauses that are true in the intended interpretation:
  • axiomatizing the domain
  - hopefully you tell it enough knowledge about the domain so that it can conclude everything you want it to
  • Ask questions about your domain

User's View of Semantics

Overview

Example Revised

\[ d = 0 \]

\[
\begin{array}{c|c|c|c}
\text{Predicates} & \text{FALSE} & \text{TRUE} \\
\hline
\text{FALSE} & \text{FALSE} & \text{TRUE} \\
\text{TRUE} & \text{FALSE} & \text{TRUE} \\
\end{array}
\]

\[
\text{Model of KB:}\ 
\begin{array}{c|c|c}
\text{KB} & \text{b} & \text{d} \\
\hline
\text{b} & \text{false} & \text{false} \\
\text{d} & \text{false} & \text{true} \\
\end{array}
\]

\[
\text{Y}: b \rightarrow d
\]
Summary of Semantics

• User has intended interpretation
• But just tells the computer a small set of facts that hopefully adequately captures the user’s intended interpretation
• Computer answers true if all interpretations that make KB true (models) make the question true
• Computer answers true if all interpretations that make KB true adequately captures the user’s intended interpretation
• But just tells the computer a small set of facts that hopefully
  Llver the intended interpretation

Computer’s view of semantics

- But number of interpretations grows quickly
- All of the possible interpretations (model checking)
  - Aside: computer could answer the question by enumerating over
    all of the possible interpretations (model checking)
    - Where computer could answer the question by narrowing down the models
      - There is at least one model in which b is false
    - Otherwise, computer could answer “I don’t know”
    - b is true in all models, so it is true in user’s intended interpretation
      - Computer should answer true if b | q
    - Llver the intended interpretation
    - Computer given the knowledge base