# Ontology 1

Assoc. Prof. Ian Watson
CS 367





## Contents







## Definition

- Ontology
  - (1): a science or study of being; specifically, a branch of metaphysics relating to the nature and relations of being. (2): a particular system according to which problems of the nature of being are investigated.
  - a theory concerning the kinds of entities and specifically the kinds of abstract entities that are to be admitted to a language system.

  *Webster's 3rd. New International Dictionary*





## Definition

- Ontology
  - (1): a science or study of being; specifically, a branch of metaphysics relating to the nature and relations of being. (2): a particular system according to which problems of the nature of being are investigated.
  - a theory concerning the kinds of entities and specifically the kinds of abstract entities that are to be admitted to a language system.
- language -> communication & understanding





## Why?

- If people or agents are to communicate they must share a common understanding
- "Waiter, that was a beautiful duck, please get me the bill"
- The elephant stood on the table, it broke!





## Why?

- Understanding means we share the meaning of words or concepts
  - Red, tiger, ocean, love, mother…
- Understanding also needs common sense
  - Which is quicker, a *fast* car or a *fast* plane?

## Why?

- Ontologies deal with defining concepts and relations
- Conceptual graphs, KIF, Ontolingua, commonKADS, ...
- They are a dictionary
- They can also be used to define problem solving K and common sense K





## Why?

- An ontology is a fomal description of the concepts and relations shared by a community of agents
- Like a formal specification of a program





## An example

- Mother
  - An animal that is the female parent of children
- Animal
  - A living thing – contains DNA
- Living
  - A temporary state for some things requiring energy
- Thing
  - An atomic concept





## An example

- Mother from the CYC ontology
  - #$biologicalMother : <#$Animal><#$FemaleAnimal> (#$biologicalMother OFFSPRING FEMALE) means that #$FemaleAnimal FEMALE is the female biological parent of the #$Animal OFFSPRING.





## An example

- #$Animal
  - The collection of all animals; this large class of organisms is one instance of #$BiologicalKingdom. Animals are typically motile, living, whole organisms; they are elements of #$Heterotroph, incapable of performing instances of #$Photosynthesis. Animal cells contain cholesterol and lack cell walls made of cellulose. #$Person is a subset of #$Animal



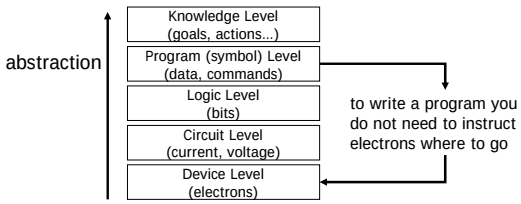


## A problem

- These definitions necessarily bottom out in expressions containing undefinable primitive atomic concepts
- We provide the *meaning*
- This is *The Chinese Room* problem identified by Searle
- The computer does not *understand* the symbols it manipulates

## The Knowledge Level

- Allen Newell (1982)
  AI Vol. 18 pp. 87-127

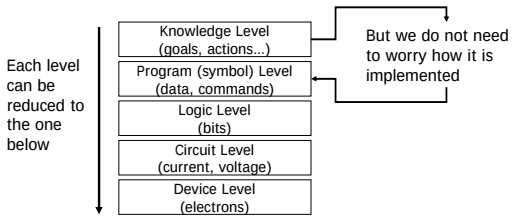







## The Knowledge Level

- To describe K you do not need to use a specific programming language








## The Knowledge Level

- Newell's principle of rationality:
  "*if an agent has knowledge that one of its actions will lead to one of its goals, then the agent will select that action*"
- A direct relation to autonomous intelligent agents
- Thus the goals and K of agents are described at the Knowledge Level





## The Knowledge Level

- Knowledge:
  "*whatever can be ascribed to an agent, such that it's behavior can be computed according to the principle of rationality*"
- K is characterised functionally in terms of what an agent does, not how it is represented (encapsulation)





## The Knowledge Level

- We require a K Level representation to describe the K and goals of our agents
- Allow our agents to communicate
- And perform according to the principle of rationality





## Declarative Programming?

- AI programs commonly contain K written as statements of fact
  - conventional programs describe procedures for manipulating data (for, until, while...)
- In Prolog
  - mother(X,Y):- female(X), parent(X,Y).
  - X is the Mother of Y if X is Female and X is the Parent of Y.

## Declarative Programming

- Prolog (Progamming in Logic) is an implementation of Predicate Calculus
- all basket ball players are tall

$$\forall X\,(\text{basketball\_player}(X) \Rightarrow X(\text{tall}))$$

- some people like brussel sprouts

$$\exists X(\text{person}(X) \Rightarrow \text{likes}(X, \text{sprouts}))$$





## Declarative Programming

- Why not use Logic as our ontological K level language?
- logic is hard to read – even for computer scientists
- Some statements are hard to express
  - "everyone loves someone"
- Poor communication medium





## K & Communication

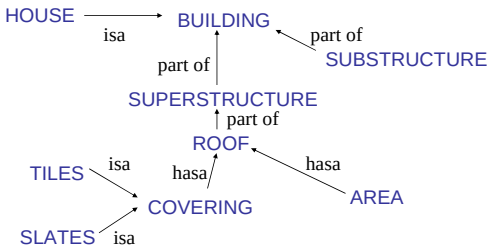
- Which is easier to understand?
  - isa(house,building).
    isa(slates,covering).
    isa(tiles,covering).
    partof(substructure,building).
    partof(superstructure,building).
    partof(roof,superstructure).
    hasa(covering,roof).
    hasa(area,roof).





## K & Communication

- Which is easier to understand?

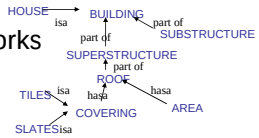







## K & Communication

- Prolog can be represented graphically
- Prolog is logic
- Therfore logic can be represented graphically
- As Semantic networks








## K & Communication

- Semantic networks range from ad hoc partially formalised representations using simple labels (isa, hasa, partof)
- To formal representations such as Conceptual Graphs

## Conceptual Graphs

- Sowa, J.F. (1984)
  *Conceptual Structures: information processing in mind and machine.*
  Addison Wesley, Reading Massachusetts
- Science Library 001.535 S73
- http://www.bestweb.net/~sowa/cg/





## Conceptual Graphs

- A logic based on existential graphs & semantic nets
  - Logically precise
  - Humanly readable
  - Direct mapping to natural language
  - Direct mapping to computer implementations
- Intermediate K Rep.





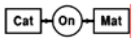
## Conceptual Graphs

- CGs have 3 forms
  - Graphical display form (DF)
    - readable
  - Linear form (LF)
    - compact
  - Conceptual graph interchange form (CGIF)
    - interoperable





## Conceptual Graphs

- A cat is on a mat
  - DF = 

  - LF = [Cat]->(On)->[Mat]
  - CGIF = (exists ((?x Cat) (?y Mat)) (On ?x ?y))
- Maps to the predicate calculus:
  $(\exists x{:}\mathrm{Cat})(\exists y{:}\mathrm{Mat})\mathrm{on}(x,y)$.





## Conceptual Graphs

- A cat is on a mat
  - A concept [Cat] 

  - A relation (On) 

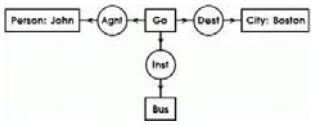
  - Relations are linked to concepts by arcs






## Conceptual Graphs

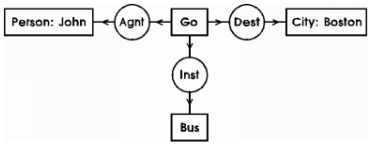
- John is going to Boston by bus




  - [Go]-
    (Agnt)->[Person: John]
    (Dest)->[City: Boston]
    (Inst)->[Bus]

## Conceptual Graphs

- John is going to Boston by bus




  - [Person: John] John is *referent*, a specific instance of a Person





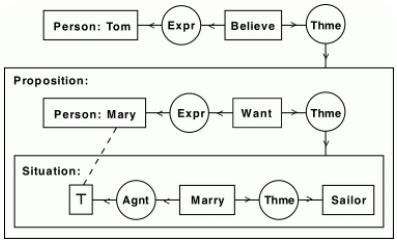
## Conceptual Graphs

- (∃x1:Person)(∃x2:Believe)(expr(x1,x2) ∧ thme(x2, (∃x3:Person)(∃x4:Want)(∃x8:Situation) (name(x3,'Mary) ∧ expr(x4,x3) ∧ thme(x4,x8) ∧ dscr(x8, (∃x5:Marry)(∃x6:Sailor) (agnt(x5,x3) ∧ thme(x5,x6))))))
- Any idea????





## Conceptual Graphs



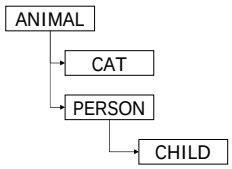

- Tom believes that Mary wants to marry a sailor





## Conceptual Graphs

- Concepts are typed (a type hierarachy)
- CHILD < PERSON *child is a person*
- PERSON < ANIMAL *person is a animal*








## Conceptual Graphs

- Concepts and relations can be described by *canonical* graphs
- ANIMAL < ANIMATE, PHYSOBJ, ¬Machine
- *an animal is an animate physical object not a machine*
- The type hierarchy & canonical graphs check that concepts and relations are used correctly
- [ACT]->(AGNT)->[ANIMATE]





## Conceptual Graphs

- TEACH < ACT
  - [TEACH] –
    (AGNT) -> [ANIMATE]
    (RCPT) -> [ANIMATE]
    (OBJ) -> [SUBJECT-MATTER]
  - [TEACH] –
    (AGNT) -> [Lecturer: Ian]
    (RCPT) -> [Student: {*}]
    (OBJ) -> [Paper: CS.367]


# Exercises

- For next lecture
  - Create some simple CGs describing going to the movies
  - Define the concepts required and the relations
  - Create type hierarchies and canonical graphs to define your concepts and realtions