Information Visualization and Knowledge Management

Dr. Margaret-Anne Storey
Dept of Computer Science, UVic

Knowledge Representation and Acquisition

Outline

- Motivation (why do we care about knowledge systems?)
- What is meant by knowledge representation?
- Parts of a knowledge system:
  - Logic
  - Ontology
  - Computation
- Knowledge representation systems
- Maintaining knowledge bases'
- Visual techniques in knowledge representation
- Challenges in knowledge representation

“Hidden Treasures”

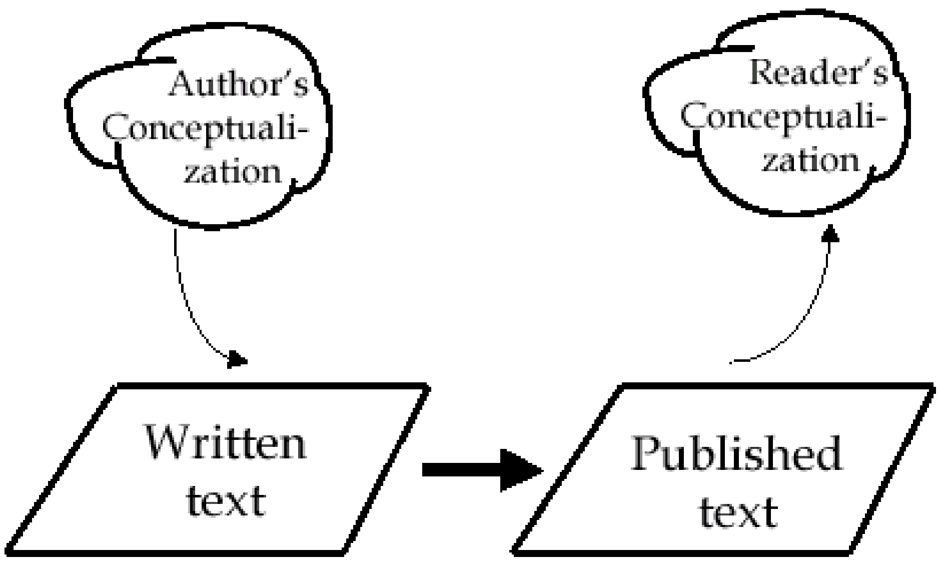
“We would be astonished at the treasures contained in our knowledge”
-- Kant

- When companies and communities begin using knowledge systems, they become more aware of how knowledge helps them do their work
- Knowledge systems can provide better methods for managing knowledge within organizations and scientific communities
- Knowledge systems can mediate communication

What is knowledge

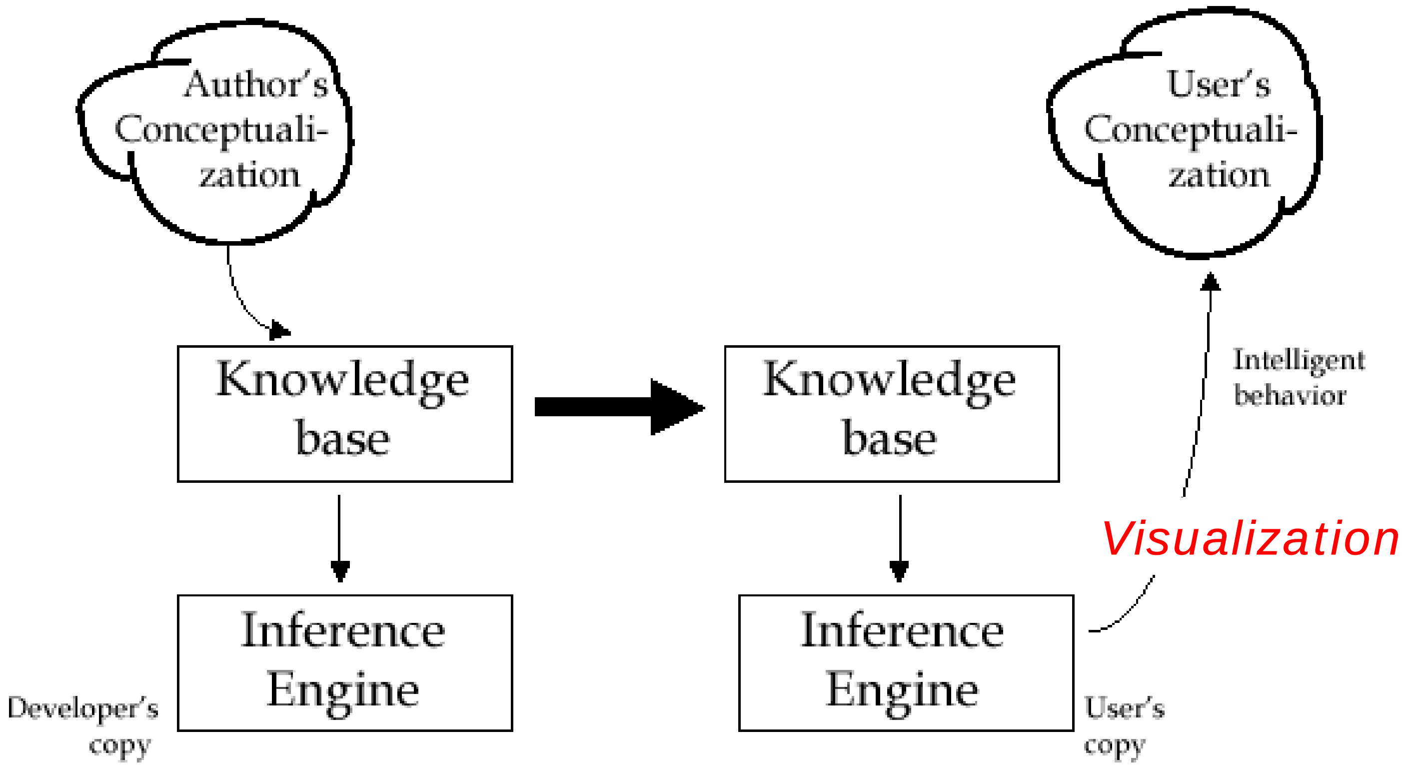
- Some definitions:
  - computable model of some domain for some purpose
  - a competence-like notion, being a potential for generating action
  - justified true belief (philosophical definition)
  - the specification of what a symbol structure should be able to do
- A knowledge representation = knowledge + access mechanism

Communication in print media



Courtesy of Mark Musen, Stanford

Communication via knowledge media



Courtesy of Mark Musen, Stanford

Knowledge systems and Computer systems

- Computer systems are embedded in larger systems
- “Hidden knowledge” is in the form of comments or in the minds of the domain experts is not really understood by the algorithms or code (semantic knowledge)
- Knowledge systems can be used by programmers and knowledge engineers to help us reflect on the hidden treasures contained in programs

Knowledge Acquisition and Modeling

- Interviews, observations of people and work patterns
- Collect knowledge about the task in terms of examples of knowledge based behaviour
- Questions to ask:
  - Who wants it?
  - What do these people do?
  - What role should the system play?
- Project team during knowledge acquisition:
  - The end user
  - The domain expert
  - The knowledge engineer
- Consider the implications of different designs (what kinds of inferences can be made and how efficiently)
- Test the system with users
- A design should be extensible, issues of maintenance

Knowledge representation (1)

- When we talk about knowledge representation we need to consider 3 things:
  - The symbols (semantics – the meaning of symbols)
  - The situation
  - An observer (i.e. an agent observes both the symbols and the situation; the symbols become a model of the situation)

  Terminology:
  - When an agent picks some set of perceived objects in the situation and makes symbols for them we say that she "reifies" them
  - When an agent associates symbols with objects in the situation, we say that the symbols "designate" the objects

Knowledge representation (2)

- A Knowledge System has 3 aspects:
  - Logic (formal structure and rules of inference)
  - Ontology (the kinds of things and individuals that exist in the application domain)
  - Computation (applications, problem solving methods)
- Knowledge representation is the application of logic and ontology to the task of constructing computable models for some domain

Logic

The very first lesson that we have a right to demand that logic shall teach us is, how to make our ideas clear; and a most important one it is, depreciated only by minds who stand in need of it. To know what we think, to be masters of our own meaning, will make a solid foundation for great and weighty thought.

-- Charles Sanders Peirce

"How to make our ideas clear"

Foundations of Logic

- Origins of documenting "knowledge"
- Aristotle's life work resulted in an encyclopaedic compilation of knowledge of his day
- Before compiling it, he had to invent words for representing his knowledge
- Terms like: category, metaphor, hypothesis, quantity, quality, genus, species, noun, verb, predicate…
- He introduced the notion of the "syllogism"
  - 3 part pattern for representing a logical deduction (Major premise, minor premise, and a conclusion)

Representation and Knowledge based systems

- Knowledge-based systems require a model of a **domain** and of a **task** to perform
- Representations enable making distinctions and inferences appropriate for relevant tasks
- Representations can differ with respect to expressiveness and/or computational complexity of answering certain queries

Trade-offs in knowledge representation

- Expressiveness vs. Tractability
  - Expressiveness measures how much you can say about the world being modeled
  - Tractability concerns knowing that the inference is decidable (will produce an answer) and will produce an answer efficiently
- Managing complexity
  - Limit what you say in a KB so that inferences are tractable
  - Accept that the knowledge will be incomplete
- First order logic is maximally expressive, but in the worst case it is undecidable (not guaranteed to produce an answer)

Varieties in logic for Knowledge Representation

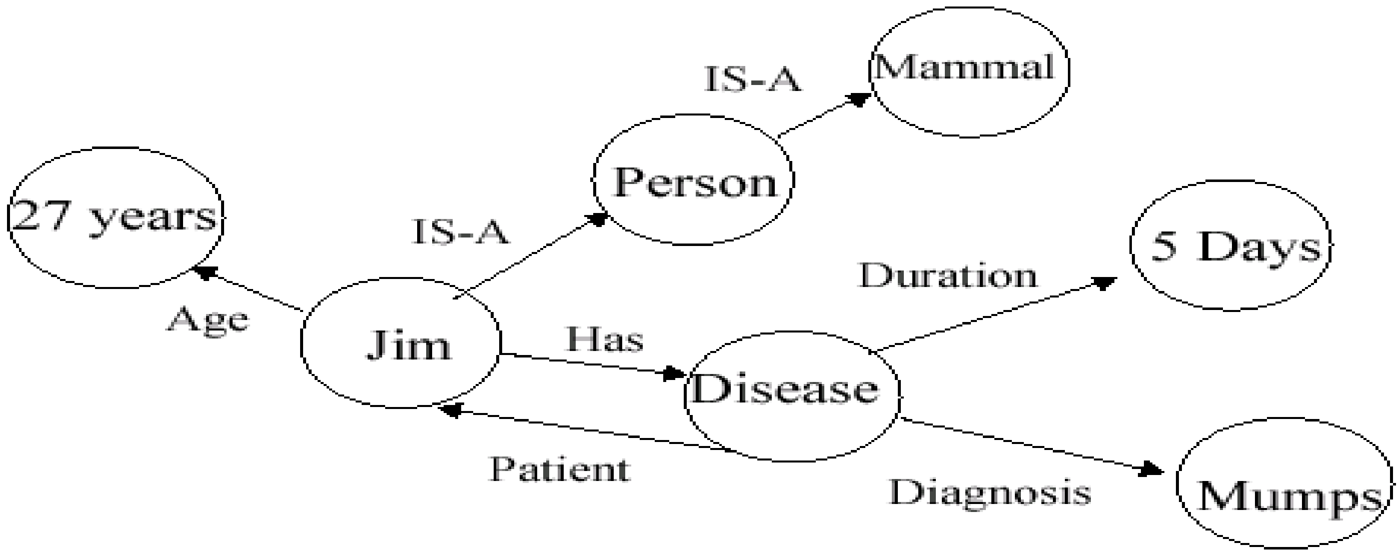
- Vocabulary (choice of symbols, language)
  - May be domain dependent or domain independent
- Syntax (grammar rules)
  - For people: differences can have a huge impact on readability, learnability, and usability
  - For computers: major impact on complexity and efficiency of theorem provers
- Semantics
  - Underlying theory
- Rules of inference

First Order Logic (FOL)

- First-order logic enjoys a special status
  - Expressive enough to define many things, including itself, computer systems, mathematics, and higher-order logics
  - Well-defined and understood model and proof theory
  - Limited set of primitives (?, ?, ?, ? …)
- The “one true logic”?
  - Some scientists have argued that only FOL is needed for KR tasks (McCarthy, Quine)
  - But there are good reasons to use other formalisms

Semantic networks

- A directed graph where Vi are concepts and Ei,j are relations



Courtesy of Mark Musen, Stanford

Semantic Networks -- Issues

- Difficult to represent N-ary relations
- Different kinds of relationships difficult to infer
- There is no real distinction between types (categories) and instances
- Many different semantic-network representations (but they agree on few things)

Other representation formalisms

- Higher/different forms of logic:
  - Non-monotonic logic
  - Modal logics (temporal, intensional)
  - Fuzzy logic
- Probabilistic representations
  - Bayesian nets
  - Neural networks
- Simulation (virtual reality)
  - Meteorology, bioinformatics

Ontologies

- Why ontological categories?
  - Logic itself has no vocabulary for describing the things that exist
- Ontologies answer the question:
  - "What is there that I need to know about?"
  - That is they help us classify things by finding ontological categories
  - May be used to maintain controlled terminologies
- Note: vocabularies provide a set of predefined terms without specifying the meaning of those terms in a precise manner (relationships among terms are often implicit)

Ontologies are used very commonly today

Ontologies are also used for…

- Mediation among heterogeneous databases
- Reference characterization of application domains (for knowledge reuse)
- Ontologies can enable:
  - Sharing of knowledge
  - Reuse of domain knowledge
  - Separation of domain knowledge from operational knowledge
  - Analysis of domain knowledge to enable decision making

Ontologies (2)

- 2 sources of ontological categories:
  - Observation (knowledge of the physical world)
  - Reasoning (to make sense of observations by generating a framework of abstractions – sometimes called metaphysics)
- Dichotomy between physical and abstract categories:
  - If a category is physical, then it will have a location in space-time
  - If a category is abstract, then it will have no physical location but it might be assigned abstract coordinates in some imaginary space
- The granularity of the categories defined depends on the reason for building the ontology

Creating Ontologies

- Manual classification:
  - Top-down
    - Starts with the goals and then pursues a framework to organize the parts in depth(philosophical approach)
    - May lack rigor
  - Bottom-up
    - Start with primitive concepts and reason about higher level abstractions
    - May lose sight of the overall goals
- Automatic classification can be done if concepts have clear definitions, classes can be organized into a hierarchy rather than the user specifying their exact place
- Hybrid classification schemes are also sometimes employed
- However concepts are not always easy to identify…

Vagueness, Uncertainty, Randomness & Ignorance

- “Knowledge Soup”
- Vague intuitions, gut feelings, …
- Knowledge chunks should be internally consistent, but often they are not
- Inconsistencies can come from:
  - Generalizations that omit “obvious” exceptions
  - Abnormal conditions
  - Incomplete definitions
  - Conflicting defaults
  - Unanticipated applications
- Languages can be a tool for discriminating and creating structure out of the knowledge soup (find little bits of order in the disorder)
- How do we know when we have the ontology “correct”?

Limitations of Logic and Ontology

- A well-debated area of KR – after certain applications that worked well in the lab failed miserably in larger, real-world situations
- Aristotle defined essential properties (humans are rational) and accidental properties (humans are featherless). But others argue these express mental categories and not reality.
- Problem: languages are discrete and well-defined, whereas the world is endless and continuous.

Some Thoughts for Discussion

- *Intertwingularity is not generally acknowledged — people keep pretending they can make things deeply hierarchical, categorizable and sequential when they can't. Everything is deeply intertwingled.*
  — Ted Nelson
- *The human mind does not work [hierarchically]. It operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. It has other characteristics, of course; trails that are not frequently followed are prone to fade, items are not fully permanent, memory is transitory*
  – Vannevar Bush
- *[there is an] esthetic or emotional need that people have for model theory. Such people take for granted that "the world splits itself up, on its own initiative, into sentence- shaped chunks". Otherwise model theory is just an exercise in translating from one language to another -*
  - Richard Rorty

Common problems in Ontology Development

- When concepts are not defined at useful levels of abstraction
- Concept definitions are overloaded (for example red bicycle)
- Hierarchical relationships are not appropriately used
- Difficult tradeoffs to consider:
  - e.g. If it is too general, it may be useless for any specific application, but if it is not general enough it may not be extensible
- Different philosophies in building ontologies:
  - The more that is modeled the better
  - Smaller customized ontologies should be created that are tailored to particular tasks

Some ontology representations

- Frame-based ontologies
- Descriptions logics

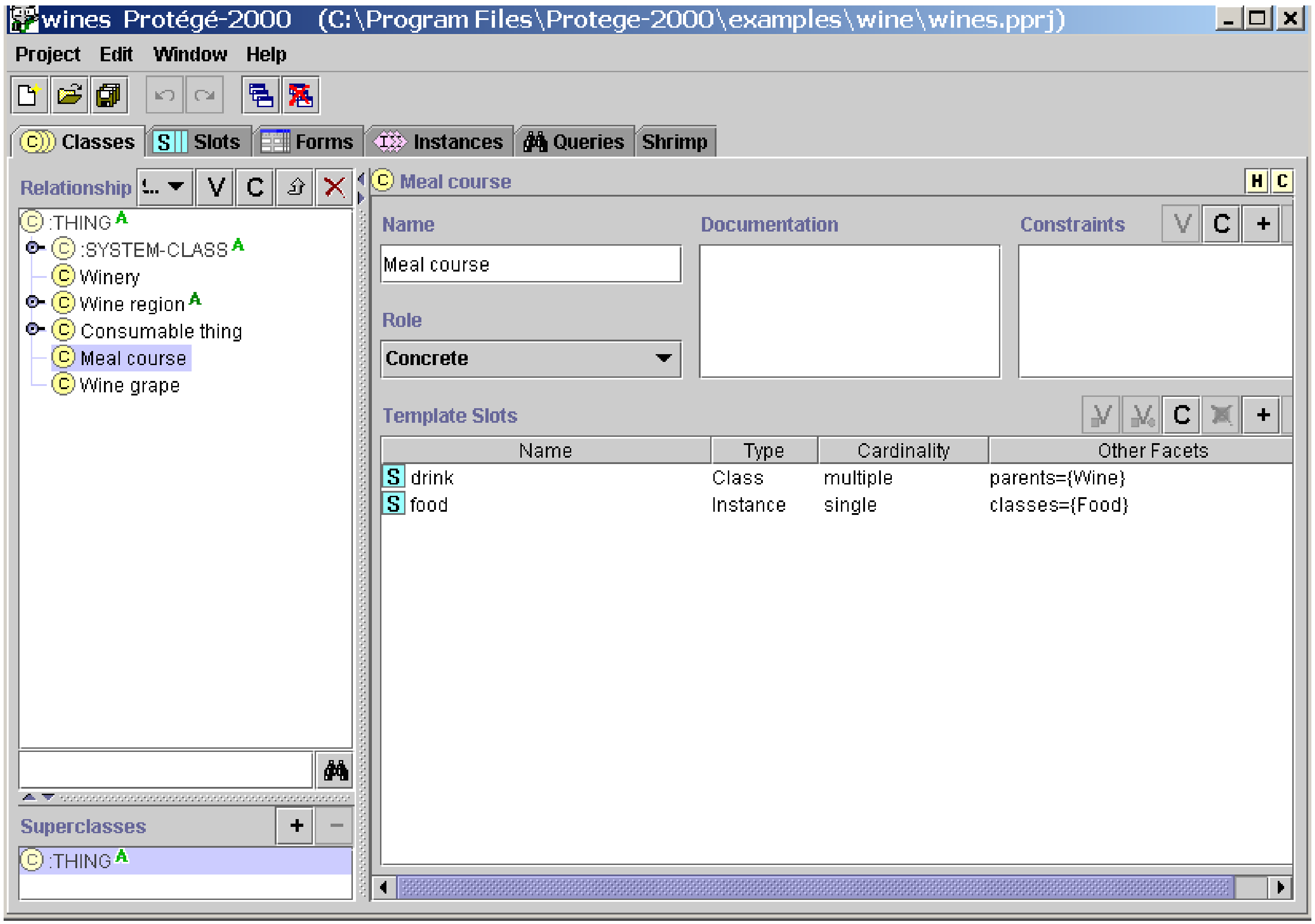
Frame–based Knowledge Representations

- A *frame* is a data structure (network of nodes and links) for representing stereotypic situations
- Attached to each frame are certain kinds of information:
  - how to use the frame
  - what is about to happen next
  - what to do if expectations are not confirmed

Frame based ontologies

- In addition to concepts (or categories) a frame based ontology enumerates:
  - attributes of concepts
  - may contain default values for some attributes
  - relationships among concepts
  - constraints on relationships among concepts
- The concepts, attributes, relationships, and constraints on relationships define a structure for the application area
- A knowledge base in a frame system has 2 parts:
  - a hierarchy of type definitions
  - A collection of instances
- Limitations: no disjunction or negation operations can be done in frame based systems

The Protégé knowledge management tool



Describing ontologies in Protégé

- Concepts in an ontology in a frame-based system are called *classes (*they describe “kinds” e.g. person)
- Subclasses inherit properties from superclasses to form a hierarchy (or heterarchy) of classes (such properties can be overridden)
- Properties of concepts are referred to as *slots* (similar to *roles* or *properties* in other systems)
- Restrictions on slots are called *facets*
- *Instances* are descriptions of individuals

✍ An ontology with a description of instances makes up a *knowledge base* in Protégé

Authoring a frame-based ontology from scratch

- Steps:
  - Define classes in the ontology (i.e. identify the main concepts in the ontology)
  - Arrange the classes in a subclass-superclass hierarchy
  - Define the slots (properties) for the classes and define the allowable values and restrictions for the slots
  - Specify the facets to provide additional descriptors or constraints for the properties (e.g. data type, default values)
  - Iterate

See Ontology 101 paper:
http://smi-web.stanford.edu/pubs/SMI_Abstracts/SMI-2001-0880.html

Advantages and Disadvantages of the frame based representation

- Advantages:
  - Frames are simple and easy to understand for humans
  - Inheritance is quite natural
  - Inference can be efficiently supported in a frame based knowledge base by following links
- Disadvantages:
  - Negation cannot be represented
  - Disjunction cannot easily be represented
  - Qualification is not part of the language (i.e. All of…)
  - Difficulty modeling n-ary relations
- Problems encountered when building frame systems if:
  - Classes are not defined at useful levels of abstraction
  - Overloading of concept definitions has been done

Description Logics

- A subset of FOL designed to focus on categories and their definitions in terms of existing relations
- Automatic classification of new concepts
- Automatic identification of redundant concepts
- Ease of customization when customers require specialized terms
- More expressive than frames and semantic networks
- Major inference tasks:
  - Subsumption (is category C2 a subset of C1)
  - Classification
- But may have end-user issues (due to automatic restructuring) and reliability concerns

KL-One

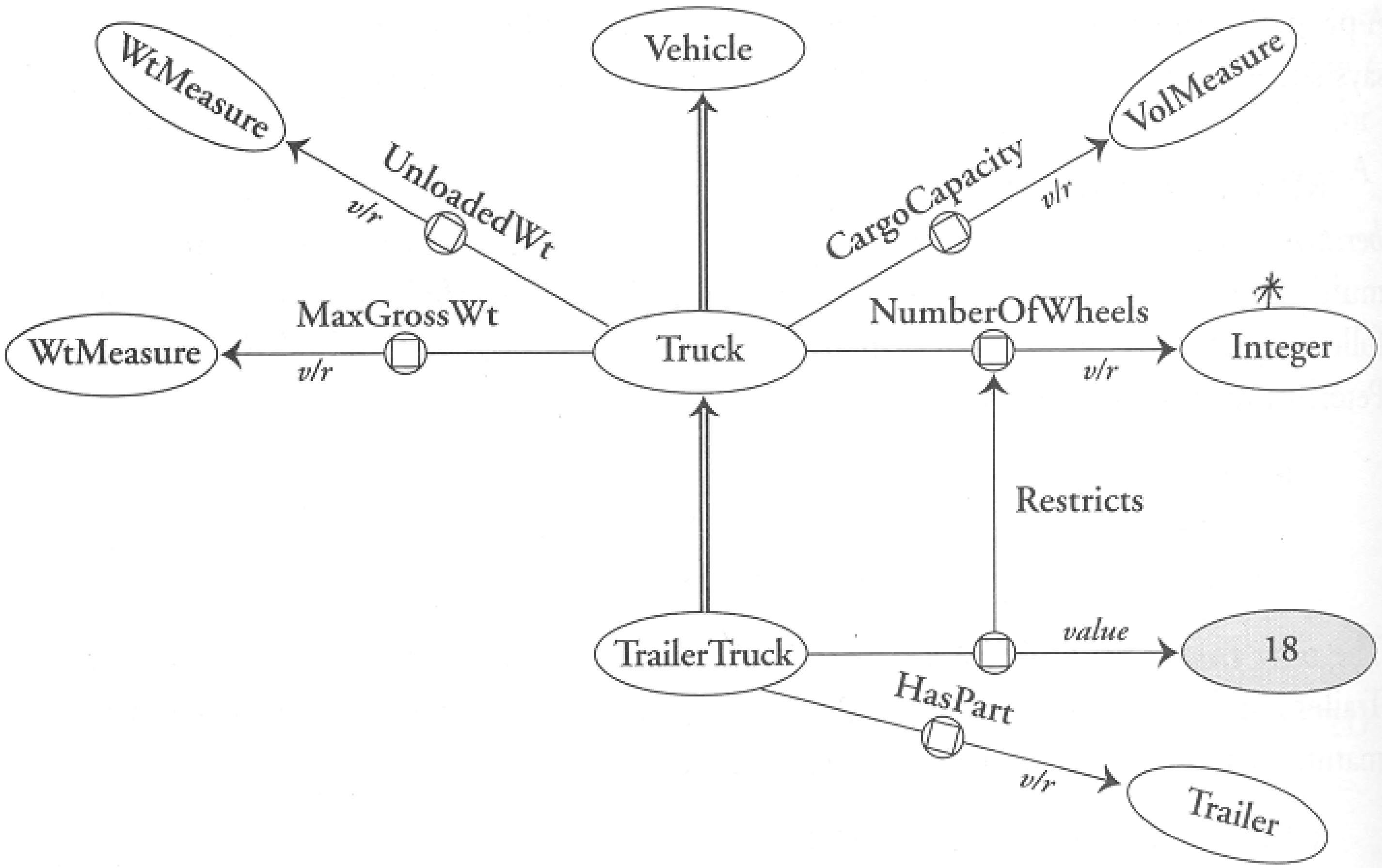


FIGURE 3.2 Truck and TrailerTruck frames in KL-ONE

Problem Solving Methods

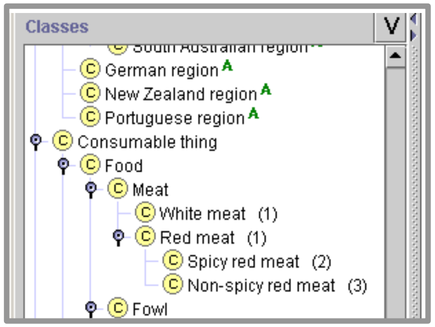
- Are knowledge systems intelligent?
  - It doesn't matter, we can still build them without considering this issue
- Many systems have explicit ontologies and PSMs
- Some examples of problem solving could include:
  - Search and optimization
  - Sequence alignment
  - Diagnostic problem solving
  - Therapeutic problem solving
  - Classification
  - Constraint satisfaction
  - Scheduling
  - Planning

Summary: Principles of Knowledge Representation

1. A knowledge representation is a surrogate for the real system that can be simulated and used for reasoning
2. A knowledge representation is a set of ontological commitments in the application domain
3. A KR is a fragmentary theory of intelligent reasoning within the application domain (as explicit axioms or as executable programs)
4. A KR is a medium for efficient computation
5. A KR is a medium for human expression facilitating communication between knowledge engineers and domain experts
   - ✍ *Our hypothesis (in this course) is that visualizations can be used to enhance this communication… we will explore this throughout the term.*

Visualization techniques in knowledge systems

- Indented lists
- Collapsible direct manipulation tree (Protégé)
- Hypertext links (Ontolingua)
- Graph representation (Ontoviz)
- SemNET – powerful techniques (3D graphics, fisheye perspective views, clustering, animation), was not widely adopted (1988)



Evolving and maintaining ontologies

- Browse and understand the existing class hierarchy in the ontology
- Browse and understand at least some of the instances, slot values and restrictions in the knowledge base
- Understand why the ontology was structured that way
- Restructure the ontology and instances, make changes
- Lack of integration can be a huge problem
- Merging ontologies (e.g. to know – savoir/connaître)
- Document decisions, findings and hypotheses

Challenges in Knowledge Management

- Knowledge bases tend to be very complex and large
- Detecting duplicate or incomplete concepts can be difficult in purely frame based systems
- Getting "quick overviews" of the size and complexity of a knowledge base can't be done with many of the tools
- Heterogeneous knowledge resources are difficult to merge (technically and cognitively)
- Collaborative authoring and maintenance
- Version control
- Different users with varying levels of expertise (knowledge engineers, domain experts, expert users, novice users, casual users)

- *Our goal? Apply visualization techniques to these problems and see if they can help*