Description Logics

Tim Finin
University of Maryland
Baltimore County

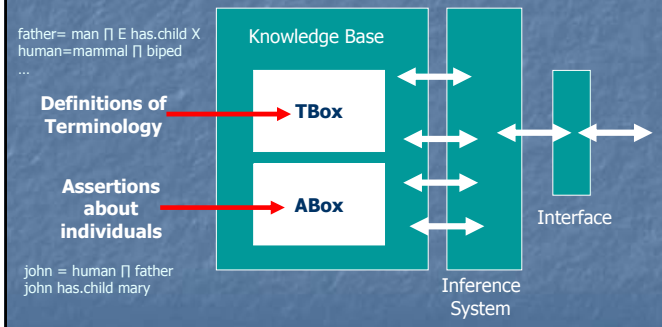
KL-ONE Style Languages

- Object-oriented representation formalisms
  http://www.ida.liu.se/labs/iislab/people/patla/DL
- Major focus of KR research in the 80's
  - Led by Ron Brachman – (AT&T Labs)
- Major systems –
  - KL-ONE, NIKL, KANDOR, BACK, CLASSIC, LOOM
- Used as the basis for the Semantic web language DAML+OIL

Description Logics

- A family of logic-based knowledge representation formalisms well-suited for the representation of and reasoning about
  - terminological knowledge
  - configurations
  - ontologies
  - database schemata
    - schema design, evolution, and query optimization
    - source integration in heterogeneous databases/data warehouses
    - conceptual modeling of multidimensional aggregation
  - descendents of semantics networks, frame-based systems, and KL-ONE
- Used as the basis for the Semantic web language DAML+OIL

Typical Architecture



DL Paradigm

- A **Description Logic** is mainly characterized by a set of constructors that allow one to build complex **concepts** and **roles** from atomic ones,
- **Concepts** correspond to classes
  - and are interpreted as sets of objects,
- **Roles** correspond to relations
  - and are interpreted as binary relations on objects

ALC

Constructor	Syntax	Example
atomic concept	A	Human
atomic role	R	likes
conjunction	C ^ D	Human ^ Male
disjunction	C V D	Nice V Rich
Negation	~C	~Meat
exists restrict.	∃R.C	∃haschild.Human
value restrict.	∀R.C	∀haschild.Blond

for concepts C and D and role R

Other constuctors

Constructor	Syntax	Example
number restriction	>= n R	>= 7 haschild
	<= n R	<= 1 hasmother
inverse role	R-	haschild-
Trans. Role	R*	haschild*
Etc.		

Intensional Description Language

- **DLs provide a composable "description language"**

```
(and Person
     (At-Least 1 Degree)
     (All Degree (One-Of BA BS MA MS PhD)))
```

  - Describes a unary predicate
  - Has variable-free syntax
- **KIF equivalent:**

```
(and (Person ?x)
     (Min-Cardinality Degree ?x 1)
     (Value-Type Degree ?x (Set-Of BA BS MA MS PhD)))
```

- **OKBC frame language equivalent:**

```
Object-Being-Described
   Instance-Of:  Person
        Degree:
             Min-Cardinality:  1
             Value-Type:  (Set-Of BA BS MA MS PhD)
```

CLASSIC Syntax

<concept> ::=
Thing | Nothing |
and (<concept> +) |
all (<role> , <concept>) |
at-least (<integer> , <role>) |
at-most (<integer> , <role>) |
same-as (<attribute-path> , <attribute-path>) |
one-of (<individual> +) |
fills (<role> , <individual> +) |
primitive (<concept> , <id>) |
disjoint-primitive (<concept> , <group-id> , <id>)

<role> ::= <identifier>

<attribute-path> ::= <identifier> | <identifier> . <attribute-path>

Example Classic Description

- CLASSIC composite description:

(and Game
(At-Least 4 Participants)
(All Participants (and Person
(Fills Gender Female))))

- OKBC frame language equivalent:

Object-Being-Described
Instance-Of: Game
Participants:
Min-Cardinality: 4
Value-Type: Female-Person
Female-Person
Subclass-Of: Person
*Gender: Female

Primitive Classes

- OKBC class frames describe *primitive* classes
  - Necessary properties of instances, e.g. –

Person
Subclass-Of: Living-Thing
*Name:
Slot-Cardinality: 1
Value-Type: String
*Child:
Value-Type: Person

- Useful for inferring -
  - Properties of an instance of a class
    (=> (C ?x) …)
  - That an object is not an instance of a class
    (=> (C ?x) (P ?x)), (not (P a)) |– (not (C a))
- Cannot infer that an object is an instance of a class

Non-Primitive Classes

- DL description language used to provide necessary and sufficient properties for class instances e.g., - University-Grad is-a

(And Person
(At-Least 1 Degree)
(All Degree (One-Of BA BS MA MS PhD)))

- KIF translation -

(<=> (University-Grad ?x)
(and (Person ?x)
(Template-Facet-Value Min-Cardinality Degree ?x 1)
(Template-Facet-Value Value-Type
Degree
?x
(Set-Of BA BS MA MS PhD))))

- Can recognize that an object is an instance of a class

Subsumption

- Subsumes ≡ Superclass-Of

  E.g., (Subsumes Person Man)
- Does C1 subsume C2?

  (=> (C2 ?x) (C1 ?x)) ?

  E.g., Game-With-At-Least-2-Participants subsumes Game-With-At-Least-4-Female-Participants

  Game-With-At-Least-4-Female-Participants is-a

  (and Game
  (At-Least 4 Participants)
  (All Participants (and Person (Fills Gender Female))))

  Game-With-At-Least-2-Participants is-a

  (and Game
  (At-Least 2 Participants))

Classification

- Determine subsumption relationships
  - For a new concept description
  - For an individual
  - See: http://www.cs.umbc.edu/771/papers/classification.mov
- Useful for -
  - Maintaining a taxonomy of concepts
  - Classifying an individual
  - Finding individuals that satisfy a description
- Subsumption in CLASSIC is
  - Polynomial and complete
  - Borgida and Patel-Schneider, J. of AI Research, v1, 1994, http://www.cs.umbc.edu/771/borgida94a.html
- Subsumption only works for non-primitive concepts
  - Problem: Most classes in most KBs are primitive(!)