Foundations of the Semantic Web
Lecture 4

More Pattern and a Problem
Classes as Values
Combining Necessary & Sufficient with Necessary conditions (General Inclusion Axioms - GICs)

Alan Rector

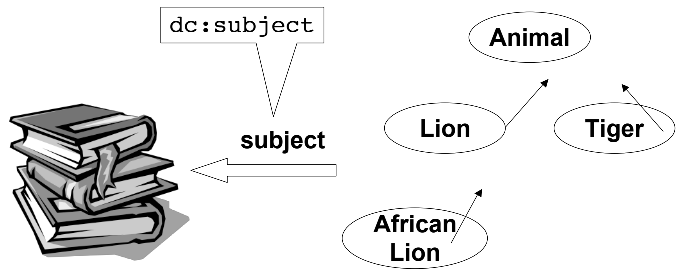
Part 1: Classes as Values

- In OWL DL nothing can be both a class and an individual
  - In classic Protégé and most frame languages everything is an individual of something
    - The class MetaClass is an instance of itself.
  - In OWL-Full a class can also be an individual
- Why the problem?
  - Paradoxes of self reference - undecidable statements are hard to avoid
    - Russell Paradox: The class of all classes that are not instances of themselves.
    - Liar Paradox (Epimenides paradox): Is the following statement true? “This statement is false”
- The logic is trickier than it looks
  - If you are interested in the theory look up Zermelo Frankel and/or Von Neuman set theory - or talk to our logician colleagues

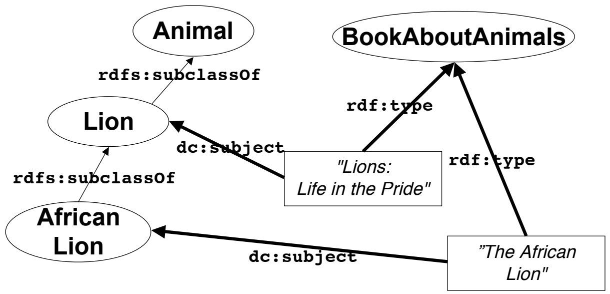
The classic application

- I want to index book / web pages / films … according to what they are ‘about’.
- The standard vocabulary for doing this in rdf is “Dublin Core” - namespace usually abbreviated to “dc:”

Using Classes as Property Values



Using Classes Directly As Values



5

Cannot do this directly in OWL 1.0

- Will use 'puns's as weak solution in OWL 1.1
- Best approximation in existing protégé
  - Name a an individual for each class with a suffix or naming-convention, e.g. lower case.
  - Provide the mirroring relation by hand or script
- There should be improved solutions coming

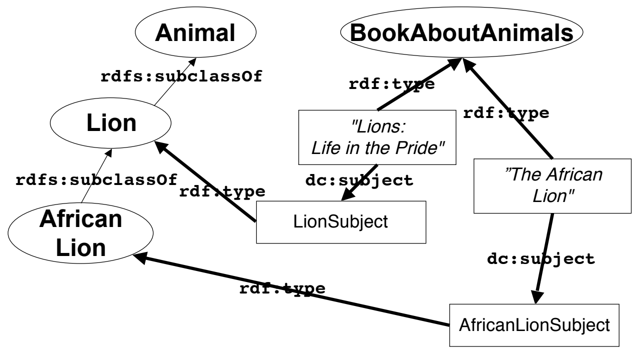
6

Approach 1: Considerations

- Compatible with OWL Full and RDF Schema
- Outside OWL DL
  - Because classes cannot be values in OWL-DL
    - Nothing can be both a class and and instance

7

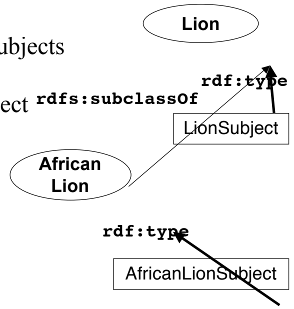
Approach 2: Hierarchy of Subjects



8

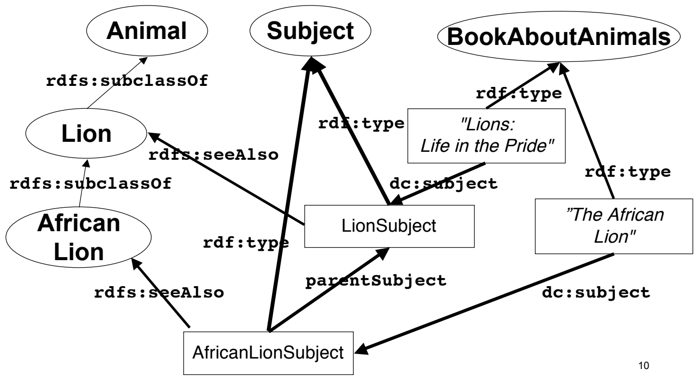
Hierarchy of Subjects: Considerations

- Compatible with OWL DL
- Instances of class Lion are now subjects
- No direct relation between LionSubject and AfricalLionSubject
- Maintenance penalty



9

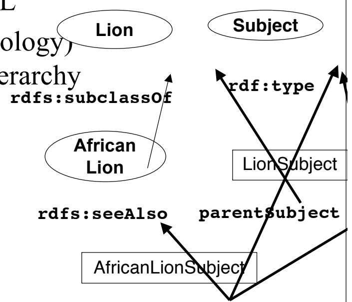
Hierarchy of Subjects



10

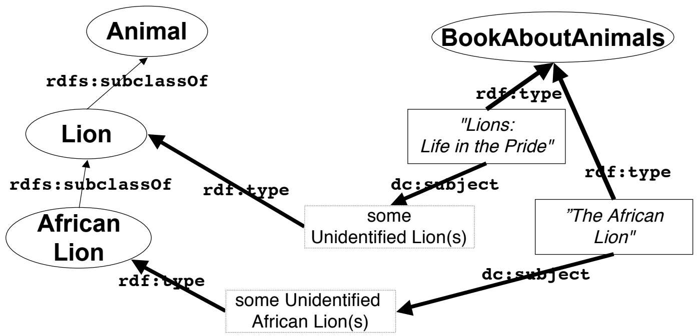
Hierarchy of Subjects: Considerations

- Compatible with OWL DL
- Subject hierarchy (terminology) is independent of class hierarchy (rdfs:seeAlso)
- Maintenance penalty



11

Apporach 3. Using members of a class as values

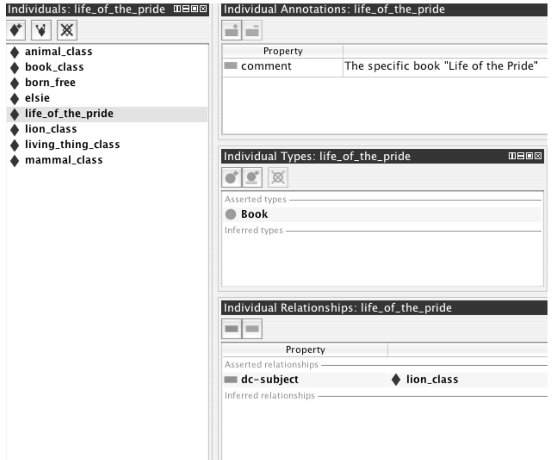


12

Protégé Examples of Approaches 2-3

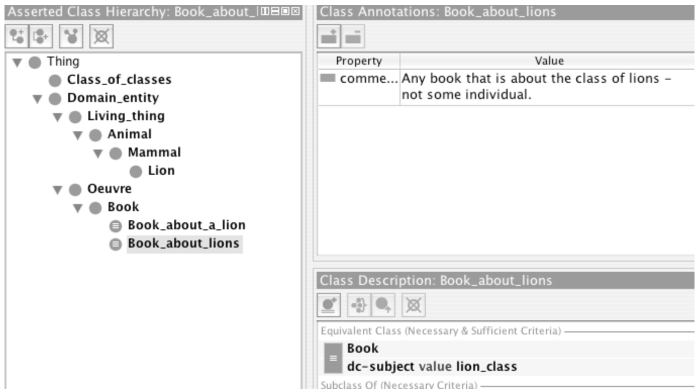
13

Representation in Protégé



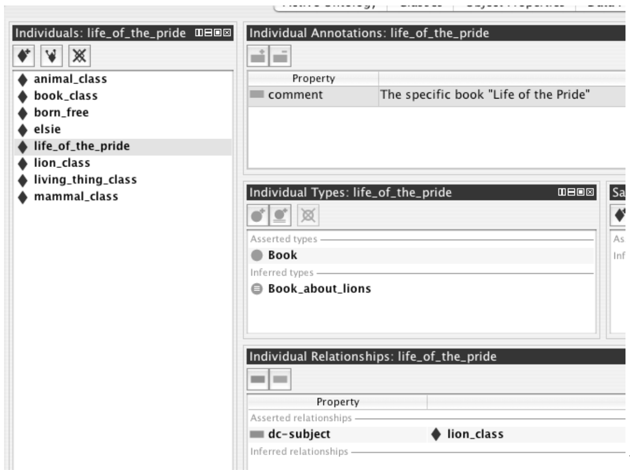
14

Class Hierarchy: Book about Lions



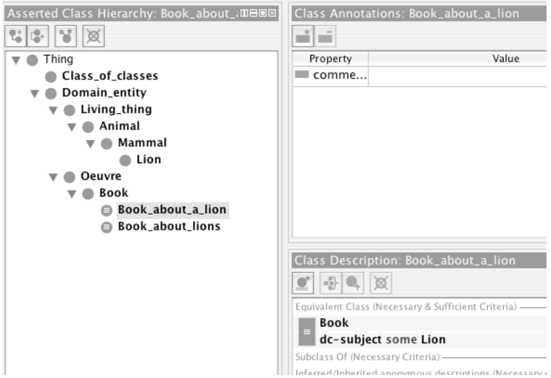
15

Inference

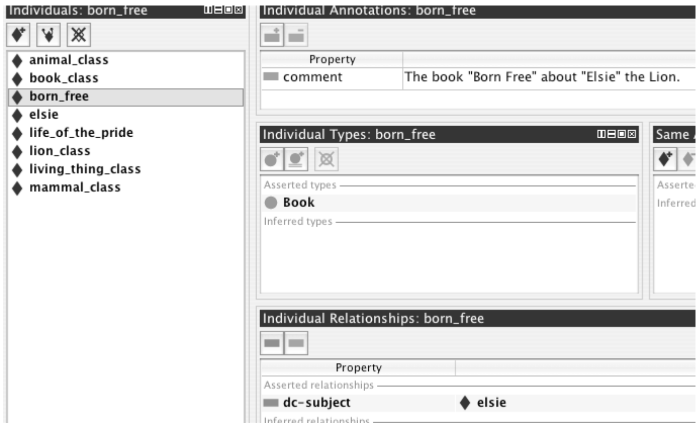


16

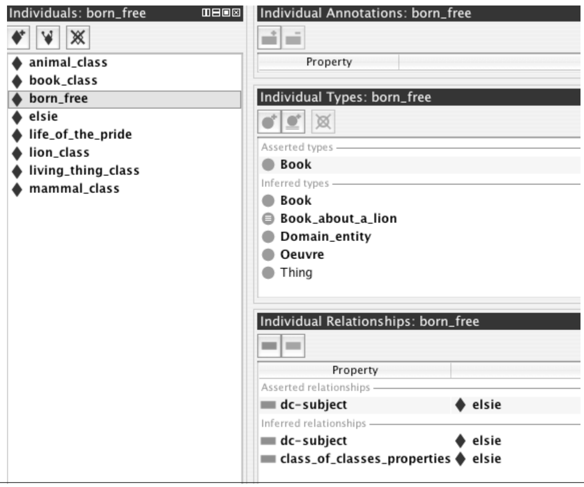
Book about SOME Lion



A Book about a Lion "Born Free"



Inference



Considerations

- Compatible with OWL DL
- Interpretation: the subject is one or more specific lions, rather than the Lion class
- Can use a DL reasoner to classify specific books
- Manchester's preferred solution
  - … but others disagree

Part 2 Defined Classes with Necessary Conditions

- OWL allows the same class to be defined and have additional necessary conditions
  - Protégé OWL has made this easy to do

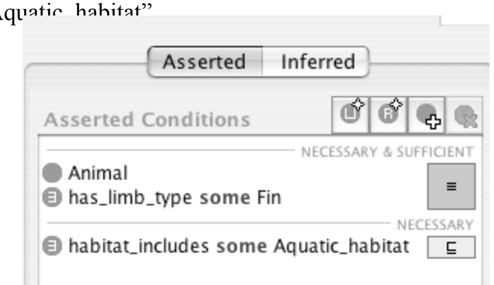
Defined classes with necessary conditions

- What does it mean to have both kinds of restrictions
  - Necessary and Sufficient
  - Necessary
- Animal_with_fins =
  Animal AND has_limb_type Fins
  ==>
  habitat_includes someValuesFrom Aquatic_habitat

Vocabulary: “Aquatic” - having to do with water.

Defined classes with necessary conditions

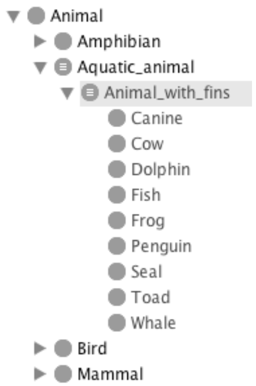
- Effectively such classes are rules or axioms
  - “Any animal with fins, has a habitat that includes includes some Aquatic_habitat”



SubclassOf means “Necessarily implies”

- Protégé OWL necessary statements necessary implications
  - Equivalent to subclass axioms
    - In fact the interface will move the class under the subclass
- Very strong statement
  - Any animal, without exception, that has fins lives in aquatic habitat
    - Think about Toads - do their habitats include aquatic?
      - The properties were phrased carefully
- Therefore defined classes with necessary statements are called “General Inclusion Axioms”
  - They are a general way of writing axioms about subsumption (“inclusion”)

Subsumption means necessary implication - the classifier produces



25

Debugging & GICs

- If a definition implies that something is classified under it that conflicts with its necessary conditions
  - The classifier will not show the classification
    - It will just show that the class is unsatisfiable but will not move it.
  - Therefore, although powerful, such constructs can be hard to debug

26

Part 3: A Ridiculously Brief Glance at Representing Time & Space

27

Extents, Intervals, and Ordering

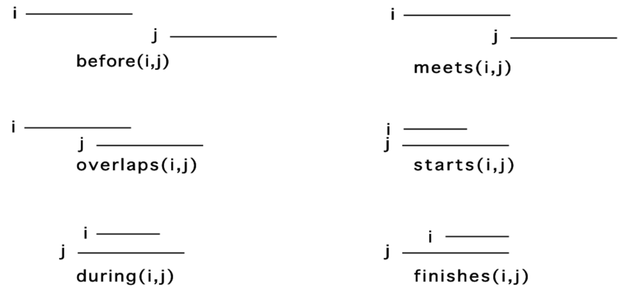
- “Extent” is a general term for a point, interval, area, volume, etc. in space and/or time
- Time comes with natural coordinates
  - Many spatial measures are also laid out with coordinates
- Timed is concerned with points and interval Space with points, intervals, areas, and volumes
- Most temporal and spatial reasoning beyond OWL

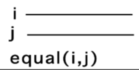
28

A few things you should know
Axioms of Ordering of time or lines

- For points in an ordered one-dimensional space
  - Anti-symmetry
    X<Y → ¬(Y<X)
  - Transitivity
    X<Y & Y<Z → X<Z
  - Totality
    X<Y v Y<X v X=Y

For an Ordered One Dimensional Space Relations between Intervals





The "Allen Calculus" specifies the results of combining intervals.
There are precisely 13 possible combinations including symmetries (6 * 2 + 1)

Exercise

- Using the diagrams on the previous slide, write down the axioms that should apply to the relations between intervals
  - r1(X,Y) & r2(Y,Z) → r?(X,Z)
    - e.g.
      before(X,Y) & before(Y,Z) → before(X,Z)

Points and Intervals

- Time representations are either point based or interval based
  - A point can be viewed as:
    - An interval of zero length
    - One of the set of ordered things that make up an interval.
  - Points can be:
    - Contained in intervals
    - The start or end of the interval
      - start(I) or end(I)

Classic Situation Calculus
Time, Situations, and Fluents

- Situation = a cross section of time
- Representation as parameter
  - "The radio was on at 9:00"
    on(radio, S9:00)
- Representation by fluents (things that can be true in situations)
  - "the radio was on at 9:00"
    true_in(s9:00, on(radio))

33

Basic Assumptions

- There is an integral measure *clock time*
  - The differential measure of clock time is *duration*
- Intervals of clock times are *sets of clock times*
  - "Kenedy was president throughout 1962"
    S ∈ year_1962 ↔ kennedy=value_in(S, president(us))
  - Intervals of clock times have *durations*

34

Events, States and Fluents

- Fluents refer to time points and may be of three types:
  - Things that can have values - *states*
    - NB "state" is used differently by other authors!
  - Things that can occur - *events*
  - Things that change things - *processes*
    - Davis defines processes as a special case of state which can be *active* or *inactive*

35

Processes and *Events*
Alternative View

- *Processes* have duration and correspond to intervals and have positive duration.
- *Events* correspond to points and have zero duration.
- *States* have values and may hold those values and have a duration but the duration may be zero.
  - In most ontologies states must correspond to intervals, though the intervals may be of zero length.

36

What is an event? A process?

- He sat down at three o'clock sharp.
- He sat down slowly and carefully.
- He was so stiff that it took him nearly a half a minute to sit down
- He sat down before the meeting.

- The birthday party took place on Tuesday
- The birthday party lasted three hours.
- The birthday party was the biggest event of the season

37

Situations and OWL/DLs

- Full situation calculus beyond OWL or DLs
  - and even to attempt it need concrete data types
- Can use the idea of a situation
  - If using an event-based view of time
    - The class of situations in which someone is sitting down at 18:00
    - Sitting_at_1800 ê
      Situation and
      (hasFluent someValuesFrom SittingProcess) and
      (occursAt someValuesFrom
      (Event and occursAt value 1800)))

38

Situations and OWL/DLs (cont)

- if using an interval based view of time
  - Sitting_between_1800_and_1801 ≡ Situation and
    (hasFluent someValuesFrom SittingProcess) and
    (occurs_during someValuesFrom
    (Interval and (hasStartTime value 1800)
    and (hasEndTime value 1801)))

39

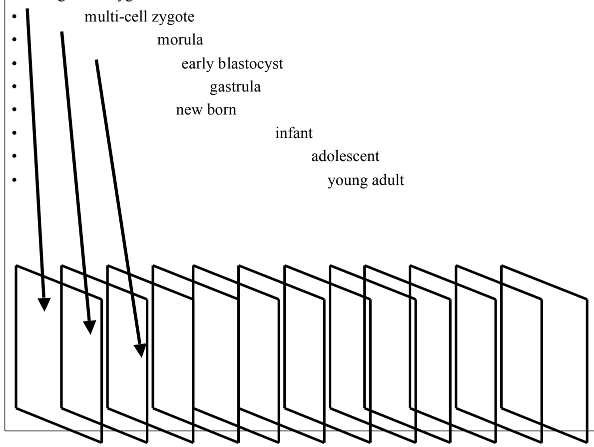
Snaps and Spans 3D and 4D views Yet another View

- Another version is to index by time
  - A "span" is entire history of an entity through time
    - Spans are intrinsically four dimensional
  - A "snap" is a cross section of a span at a point in time.
    - Qualities of continuants are dependent on the SNAP and change in the course of a SPAN
      - e.g. an Apple can be green in one SNAP and red in a later SNAP
  - A "situation" is a piece of situated information in a 4-D universe; a "Snap" is a three D section of a 4 d entity

Due to Barry Smith et al
(google "Barry Smith")

40

- single-cell zygote
- multi-cell zygote
- morula
- early blastocyst
- gastrula
- new born
- infant
- adolescent
- young adult



The Future of Time in OWL

- Might also represent ordering of time or intervals, but
  - most useful applications require both concrete domains and individuals
  - highly speculative at this time
  - but description logics are closely related to formally to temporal logics, so …