

# INFS3101/7100 Ontology and the Semantic Web

## Week 11 Tutorial OWL

Semester 1, 2006

## Key Terms

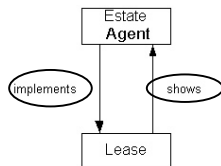
- OWL extends RDFS
- Object and datatype properties
- Universal class Thing
- Restrictions are subsets of property domains
- Names not satisfy unique name assumption
- Class descriptions can be enumeration or boolean combination of classes
- Ontology Properties
- OWL Full, OWL DL, OWL Lite

**Q1A** Elaborate a significant portion of the ontology in OWL using the OWL facilities from the lecture

### • Simple Properties

#### – owl:ObjectProperty

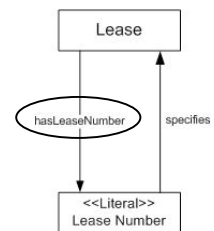
```
<owl:ObjectProperty rdf:ID="implements">
  <rdfs:domain rdf:resource="#Estate Agent"/>
  <rdfs:range rdf:resource="#Lease"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="shows">
  <rdfs:domain rdf:resource="#Lease"/>
  <rdfs:range rdf:resource="#Estate Agent"/>
</owl:ObjectProperty>
```



## Q1A: Simple Properties

### • owl:DatatypeProperty

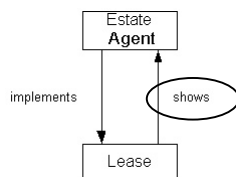
```
<owl:Class rdf:ID="Lease" />
<owl:DatatypeProperty rdf:ID="hasLeaseNumber">
  <rdfs:domain rdf:resource="#Lease" />
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>
```



## Q1A: Property Characteristics

- owl:FunctionalProperty
- one-to-one relationship

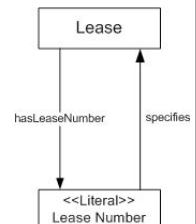
```
<owl:Class rdf:ID="Lease"/>
<owl:ObjectProperty rdf:ID="shows">
  <rdf:type rdf:resource="&owl;FunctionalProperty" />
  <rdfs:domain rdf:resource="#Lease" />
  <rdfs:range rdf:resource="#Estate A" />
</owl:ObjectProperty>
```



## Q1A: Property Characteristics

### • owl:InverseOf

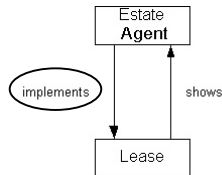
```
<owl:ObjectProperty rdf:ID="hasLeaseNumber">
  <rdf:type rdf:resource="&owl;FunctionalProperty" />
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="specifies">
  <rdf:type rdf:resource="&owl;FunctionalProperty" />
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="specifies">
  <owl:inverseOf rdf:resource="#hasLeaseNumber" />
</owl:ObjectProperty>
```



## Q1A: Property Characteristics

- owl:InverseFunctionalProperty
- It is the inverse properties of function properties
- In this case, the inverse function property implements is one-to-many relationship

```
<owl:ObjectProperty rdf:ID="shows" />
<owl:ObjectProperty rdf:ID="implements">
  <rdf:type rdf:resource="&owl:InverseFunctionalProperty" />
  <owl:inverseOf rdf:resource="shows" />
</owl:ObjectProperty>
```



## Q1A: Property Characteristics

- owl:SymmetricProperty
- Mary and John, both of whom are lessees, are sharing a house. We can say either Mary is the housemate of John, or John is the housemate of Mary.

```
<owl:ObjectProperty rdf:ID="housemate">
  <rdf:type rdf:resource="&owl:SymmetricProperty" />
  <rdfs:domain rdf:resource="Lessee" />
  <rdfs:range rdf:resource="Lessee" />
</owl:ObjectProperty>
<Lessee rdf:ID="Mary">
  <housemate rdf:resource="John" />
</Lessee>
```

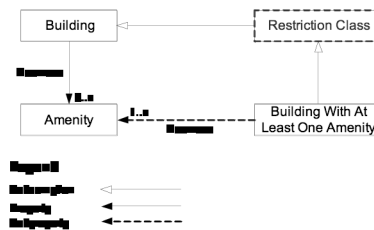
## Q1A: Property Characteristics

- owl:TransitiveProperty
- A particular clause such as rent consists of a number of subclauses such as rent in advance, in turn; the rent in advance also consists of its subclauses such as for a period agreement – 2 weeks rent.

```
<owl:ObjectProperty rdf:ID="consistsOf">
  <rdf:type rdf:resource="&owl:TransitiveProperty" />
  <rdfs:domain rdf:resource="Lease" />
  <rdfs:range rdf:resource="Clause" />
</owl:ObjectProperty>
<Clause rdf:ID="rent">
  <consistsOf rdf:resource="rent_in_advance" />
</Clause>
<Clause rdf:ID="rent_in_advance">
  <consistsOf rdf:resource="for_a_period_agreement-2-weeks_rent" />
</Clause >
```

## Q1A: Property Restrictions

- owl:minCardinality
  - There could be some of instances of *nearness* domain *Building*, which have at least one instance of *Amenity*.



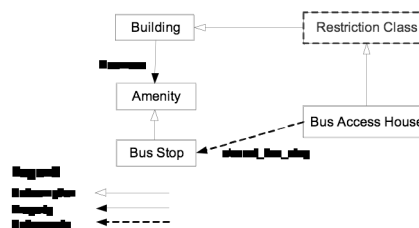
## Q1A: Property Restrictions

- owl:minCardinality
 

```
<owl:ObjectProperty rdf:ID="nearness">
  <rdfs:domain rdf:resource="Building" />
  <rdfs:range rdf:resource="Amenity" />
</owl:ObjectProperty>
<owl:Class rdf:ID="Restriction_Class">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="nearness" />
      <owl:minCardinality
        rdf:datatype="xsd:nonNegativeInteger">1
      </owl:minCardinality>
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="Restriction_Class">
  <rdfs:subClassOf rdf:resource="Building" />
</owl:Class>
```

## Q1A: Property Restrictions

- owl:allValuesFrom
- There could be a subset of instances of *Building*, which are close to bus stops alone (from a single property)



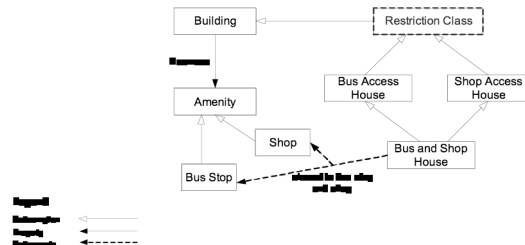
## Q1A: Property Restrictions

- owl:allValuesFrom
 

```
<owl:ObjectProperty rdf:ID="closest_bus_stop">
  <rdfs:subPropertyOf rdf:resource="#nearness" />
  <rdfs:range rdf:resource="#Bus_Stop" />
</owl:ObjectProperty>
<owl:Class rdf:ID="Restriction_Class">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#closest_bus_stop" />
      <owl:allValuesFrom rdf:resource="#Bus_Stop" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID="
  " Restriction_Class_Building_close_to_Bus_Stop">
  <rdfs:subClassOf rdf:resource="#Building"/>
</owl:Class>
```

## Q1A: Property Restrictions

- owl:someValuesFrom
- There are instances of building, which are close to at least bus stops.



## Q1A: Property Restrictions

- owl:someValuesFrom
 

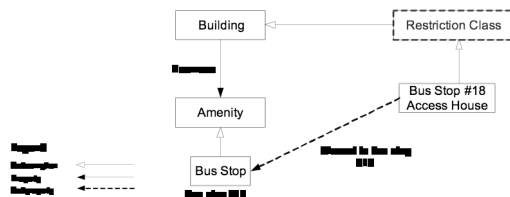
```
<owl:Class rdf:ID="Bus_Stop_and_Shop">
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Bus_Stop" />
    <owl:Class rdf:about="#Shop" />
  </owl:unionOf>
</owl:Class>
<owl:ObjectProperty rdf:ID="closest_bus_stop_and_shop">
  <rdfs:subPropertyOf rdf:resource="#nearness" />
  <rdfs:range rdf:resource="#Bus_Stop_and_Shop" />
</owl:ObjectProperty>
<owl:Class rdf:ID="Restriction_Class">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#closest_bus_stop_and_shop"/>
      <owl:someValuesFrom rdf:resource="#Bus_Stop" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
```

## Q1A: Property Restrictions

- ```
<owl:Class rdf:ID="Restriction_Class">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty
        rdf:resource="#closest_bus_stop_and_shop"/>
      <owl:someValuesFrom rdf:resource="#Shop" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID=" Restriction_Class">
  <rdfs:subClassOf rdf:resource="#Building"/>
</owl:Class>
<owl:Class rdf:ID=" Restriction_Class">
  <rdfs:subClassOf rdf:resource="#Building"/>
</owl:Class>
```

## Q1A: Property Restrictions

- owl:hasValue
- There are instances of building which are only close to a particular bus stop such as bus stop number 18 (a particular instance of the property's range)



## Q1A: Property Restrictions

- owl:hasValue
 

```
<owl:ObjectProperty rdf:ID="closest_bus_stop_18">
  <rdfs:subPropertyOf rdf:resource="#nearness" />
  <rdfs:range rdf:resource="#Bus_Stop" />
</owl:ObjectProperty>
<owl:Class rdf:ID="Restriction_Class_Building_close_to_Bus_Stop_18">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#closest_bus_stop_18" />
      <owl:hasValue rdf:resource="#bus_stop#18" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
<owl:Class rdf:ID=" Restriction_Class_Building_close_to_Bus_Stop_18">
  <rdfs:subClassOf rdf:resource="#Building"/>
</owl:Class>
```

## Q1A: Identity Between Individuals

- owl:sameAs
- Names not satisfy unique name assumption in OWL
- Two individuals may be stated to be the same in OWL. For example, the name, e.g. Ray White, and ABN such as 490589675432, of an estate agent refer to the same object.

```
<Estate_Agent rdf:ID="Ray_White">
  <owl:sameAs rdf:resource="#ABN_490589675432" />
</Estate_Agent>
```

## Q1A: Different Individuals

- owl:differentFrom - for example, Mary, a lessee is different from John who is a lessee as well.
- owl:allDifferent - for example, Smith, Jane and Bill, all of whom are lessor, are different.

```
<Lessee rdf:ID="Mary">
  <owl:differentFrom rdf:resource="#John"/>
</Lessee>

<owl:AllDifferent>
  <owl:distinctMembers rdf:parseType="Collection">
    <rental:Lessor rdf:about="#Smith" />
    <rental:Lessor rdf:about="#Jane" />
    <rental:Lessor rdf:about="#Bill" />
  </owl:distinctMembers>
</owl:AllDifferent>
```

## Q1A: Enumerated Classes

- Regarding to oneOf, for example, there are eight states in Australia: they are QLD, NSW, ACT, TAS, VIC, SA, WA and NT.

```
<owl:Class rdf:ID="States_in_Australia">
  <owl:oneOf rdf:parseType="Collection">
    <owl:States_in_Australia rdf:about="#QLD"/>
    <owl:States_in_Australia rdf:about="#NSW"/>
    <owl:States_in_Australia rdf:about="#ACT"/>
    <owl:States_in_Australia rdf:about="#TAS"/>
    <owl:States_in_Australia rdf:about="#VIC"/>
    <owl:States_in_Australia rdf:about="#SA"/>
    <owl:States_in_Australia rdf:about="#WA"/>
    <owl:States_in_Australia rdf:about="#NT"/>
  </owl:oneOf>
</owl:Class>
```

## Q1A: Set Operators

- OWL DL and OWL Full allow arbitrary Boolean combinations of classes and restrictions
- owl:IntersectionOf and owl:ComplementOf, for example, there are units except 2-bedroom ones.

```
<owl:Class rdf:ID="Non_2-Bedroom_Unit">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Unit"/>
  </owl:intersectionOf>
  <owl:Class>
    <owl:complementOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#has" />
        <owl:hasValue rdf:resource="#2_bedroom" />
      </owl:Restriction>
    </owl:complementOf>
  </owl:Class>
</owl:Class>
```

## Q1A: Set Operators

- owl:UnionOf, for example, find the leases that are implemented by either Ray White or PRD.

```
<owl:Class rdf:ID="Wanted_Lease">
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Lease_implemented_by_Ray_White"/>
    <owl:Class rdf:about="#Lease_implemented_by_PRD" />
  </owl:unionOf>
</owl:Class>
```

- Notes

- A intersecting B is a subclass of A and B
- Both A and B are subclasses of A union B.

## Q1A: Disjoint Classes

- Regarding to owl:disjointWith, for units, there are studios, one-bedroom units and 2-bedroom units, all of which are disjoint.

```
rental:One-Bedroom_Unit rdfs:subClassOf rental:Unit
rental:Two-Bedroom_Unit rdfs:subClassOf rental:Unit
<owl:Class rdf:ID="One-Bedroom_Unit">
  <rdfs:subClassOf rdf:resource="#Unit"/>
</owl:Class>
<owl:Class rdf:ID="Two-Bedroom_Unit">
  <rdfs:subClassOf rdf:resource="#Unit"/>
</owl:Class>
<owl:Class rdf:ID="One-Bedroom_Unit">
  <owl:disjointWith rdf:resource="#Two-Bedroom_Unit"/>
</owl:Class>
```

## Consultation Sessions

- Today 2-3 at 78-631