

# Knowledge Representation and Reasoning Homework 2

1. Give a translation in OWL of the TBOX above.

The owl was generated using **Protege 5.2.0**.

The owl file could also be found at the address [0]

- **has-part** is transitive and its inverse is **is-part-of**
- **is-managed-by** - its inverse is **manages**. You must specify also that **is-managed-by** is a functional role, i.e., an entity cannot be 'managed by' two different entities.
- **employs** - its inverse is **is-employed-by**.

```
-<rdf:RDF xml:base="urn:absolute:enterprise.owl">
  <owl:Ontology rdf:about="urn:absolute:enterprise.owl"/>
-<!--
  //////////////////////////////////////
  //
  // Object Properties
  //
  //////////////////////////////////////

  -->
  <!-- urn:absolute:enterprise.owl#employs -->
-<owl:ObjectProperty rdf:about="urn:absolute:enterprise.owl#employs">
  <owl:inverseOf rdf:resource="urn:absolute:enterprise.owl#is-employed-by"/>
</owl:ObjectProperty>
  <!-- urn:absolute:enterprise.owl#has-part -->
-<owl:ObjectProperty rdf:about="urn:absolute:enterprise.owl#has-part">
  <owl:inverseOf rdf:resource="urn:absolute:enterprise.owl#is-part-of"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#TransitiveProperty"/>
</owl:ObjectProperty>
  <!-- urn:absolute:enterprise.owl#is-employed-by -->
<owl:ObjectProperty rdf:about="urn:absolute:enterprise.owl#is-employed-by"/>
  <!-- urn:absolute:enterprise.owl#is-managed-by -->
-<owl:ObjectProperty rdf:about="urn:absolute:enterprise.owl#is-managed-by">
  <owl:inverseOf rdf:resource="urn:absolute:enterprise.owl#manages"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:range rdf:resource="urn:absolute:enterprise.owl#Employee"/>
</owl:ObjectProperty>
  <!-- urn:absolute:enterprise.owl#is-part-of -->
<owl:ObjectProperty rdf:about="urn:absolute:enterprise.owl#is-part-of"/>
  <!-- urn:absolute:enterprise.owl#manages -->
-<owl:ObjectProperty rdf:about="urn:absolute:enterprise.owl#manages">
  <rdfs:domain rdf:resource="urn:absolute:enterprise.owl#Employee"/>
</owl:ObjectProperty>
```

There could also be added **range Employee** for **is-managed-by** or for **employs** the domain **Someone**, but is not specified in the enunciation of the problem.

- An enterprise is managed by someone and employs someone.

```
<!-- urn:absolute:enterprise.owl#Enterprise -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Enterprise">
  - <rdfs:subClassOf>
    - <owl:Class>
      - <owl:intersectionOf rdf:parseType="Collection">
        - <owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#employs"/>
          <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Someone"/>
        </owl:Restriction>
        - <owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-managed-by"/>
          <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Someone"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
```

- A department is a part of an enterprise.

```
<!-- urn:absolute:enterprise.owl#Department -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Department">
  - <rdfs:subClassOf>
    - <owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-part-of"/>
      <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

- An office is a part of a department.

```
<!-- urn:absolute:enterprise.owl#Office -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Office">
  - <rdfs:subClassOf>
    - <owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-part-of"/>
      <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Department"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```



- The departments are exactly: Production, Research, Administration, Trade, HumanResources, PublicRelations.

```

<!-- urn:absolute:enterprise.owl#HumanResources -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#HumanResources">
  <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Department"/>
</owl:Class>

<!-- urn:absolute:enterprise.owl#Production -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Production">
  <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Department"/>
</owl:Class>

<!-- urn:absolute:enterprise.owl#PublicRelations -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#PublicRelations">
  <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Department"/>
</owl:Class>

<!-- urn:absolute:enterprise.owl#Research -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Research">
  <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Department"/>
</owl:Class>

<!-- urn:absolute:enterprise.owl#Trade -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Trade">
  <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Department"/>
</owl:Class>

<!-- urn:absolute:enterprise.owl#Administration -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Administration">
  <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Department"/>
</owl:Class>

```

- An employee is someone who is employed by an enterprise or by some part of an enterprise.

```

<!-- urn:absolute:enterprise.owl#Employee -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Employee">
  - <owl:equivalentClass>
    - <owl:Class>
      - <owl:unionOf rdf:parseType="Collection">
        - <owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-employed-by"/>
          <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
        </owl:Restriction>
        - <owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-employed-by"/>
          - <owl:someValuesFrom>
            - <owl:Restriction>
              <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-part-of"/>
              <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
            </owl:Restriction>
            </owl:someValuesFrom>
          </owl:Restriction>
        </owl:unionOf>
      </owl:Class>
    </owl:equivalentClass>
    <rdf:subClassOf rdf:resource="urn:absolute:enterprise.owl#Someone"/>
  </owl:Class>

```

- An administrative-employee is someone who is employed by an administration department or by some part of an administration department.

```

<!-- urn:absolute:enterprise.owl#Administrative-Employee -->
-->
<owl:Class rdf:about="urn:absolute:enterprise.owl#Administrative-Employee">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-employed-by"/>
          <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Administration"/>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-employed-by"/>
          <owl:someValuesFrom>
            <owl:Restriction>
              <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#is-part-of"/>
              <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Administration"/>
            </owl:Restriction>
            </owl:someValuesFrom>
          </owl:Restriction>
        </owl:unionOf>
      </owl:Class>
    </owl:equivalentClass>
    <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Employee"/>
  </owl:Class>

```

- If someone manages some entity he is an employee.  
This can be seen in the property for manages → the domain is Employee.

- A high-tech enterprise is an enterprise which has a research department.

```

<!-- urn:absolute:enterprise.owl#HighTechEnterprise -->
-->
<owl:Class rdf:about="urn:absolute:enterprise.owl#HighTechEnterprise">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#has-part"/>
      <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Research"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
</owl:Class>

```

- An industrial enterprise is an enterprise which has a production department and has at least 100 employees.

```
<!-- urn:absolute:enterprise.owl#IndustrialEnterprise -->
-<owl:Class rdf:about="urn:absolute:enterprise.owl#IndustrialEnterprise">
  -<owl:equivalentClass>
    -<owl:Class>
      -<owl:intersectionOf rdf:parseType="Collection">
        -<owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#has-part"/>
          <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Production"/>
        </owl:Restriction>
        -<owl:Restriction>
          <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#employs"/>
          <owl:minQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">100</owl:minQualifiedCardinality>
          <owl:onClass rdf:resource="urn:absolute:enterprise.owl#Employee"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
</owl:Class>
```

- A small enterprise is an enterprise which employs at most 20 employees.

```
<!-- urn:absolute:enterprise.owl#SmallEnterprise -->
-<owl:Class rdf:about="urn:absolute:enterprise.owl#SmallEnterprise">
  -<owl:equivalentClass>
    -<owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#employs"/>
      <owl:maxQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">20</owl:maxQualifiedCardinality>
      <owl:onClass rdf:resource="urn:absolute:enterprise.owl#Employee"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
</owl:Class>
```

- A big enterprise is an enterprise which employs at least 80 employees.

```
<!-- urn:absolute:enterprise.owl#BigEnterprise -->
-<owl:Class rdf:about="urn:absolute:enterprise.owl#BigEnterprise">
  -<owl:equivalentClass>
    -<owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#employs"/>
      <owl:minQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">80</owl:minQualifiedCardinality>
      <owl:onClass rdf:resource="urn:absolute:enterprise.owl#Employee"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
</owl:Class>
```

- A family-based enterprise is an enterprise with at most 4 employees.

```
<!-- urn:absolute:enterprise.owl#FamilyBasedEnterprise -->
-<owl:Class rdf:about="urn:absolute:enterprise.owl#FamilyBasedEnterprise">
  -<owl:equivalentClass>
    -<owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#employs"/>
      <owl:maxQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">4</owl:maxQualifiedCardinality>
      <owl:onClass rdf:resource="urn:absolute:enterprise.owl#Employee"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Enterprise"/>
</owl:Class>
```



- A top manager is someone who manages a big enterprise.

```
<!-- urn:absolute:enterprise.owl#TopManager -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#TopManager">
  - <owl:equivalentClass>
    - <owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#manages"/>
      <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#BigEnterprise"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Employee"/>
</owl:Class>
```

- A manager is someone who manages a department.

```
<!-- urn:absolute:enterprise.owl#Manager -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Manager">
  - <owl:equivalentClass>
    - <owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#manages"/>
      <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Department"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Employee"/>
</owl:Class>
```

- A boss is someone who manages an office.

```
<!-- urn:absolute:enterprise.owl#Boss -->
- <owl:Class rdf:about="urn:absolute:enterprise.owl#Boss">
  - <owl:equivalentClass>
    - <owl:Restriction>
      <owl:onProperty rdf:resource="urn:absolute:enterprise.owl#manages"/>
      <owl:someValuesFrom rdf:resource="urn:absolute:enterprise.owl#Office"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="urn:absolute:enterprise.owl#Employee"/>
</owl:Class>
```

2. Give a translation in Description Logic of both the TBOX and the ABOX above.

TBOX:

- An enterprise is managed by someone and employs someone.

Enterprise = Managed.Someone  $\wedge$  Employs.Someone (Someone represents all the employees)

- A department is a part of an enterprise.
- The departments are exactly: Production, Research, Administration, Trade, HumanResources, PublicRelations.

Department = isPartOf.Enterprise  $\wedge$  (hasDepartmentType.Production  $\vee$  hasDepartmentType.Research  $\vee$  hasDepartmentType.Administration  $\vee$  hasDepartmentType.HumanResources  $\vee$  hasDepartmentType.PublicRelations)

- An office is a part of a department  
Office = isPartOf.Department

- An employee is someone who is employed by an enterprise or by some part of an enterprise
- If someone manages some entity he is an employee.

Employee = Someone  $\wedge$  (isEmployed.Enterprise  $\vee$  isEmployed.Department  $\wedge$  isEmployed.Office  $\vee$  Manages.(Enterprise  $\cup$  Department  $\cup$  Office))

- An administrative-employee is someone who is employed by an administration department or by some part of an administration department.

AdministrativeEmployee = Someone  $\wedge$  ((isEmployed.Department  $\wedge$  isDepartmentType.Administration)  $\vee$  (isEmployed.Office  $\wedge$  isPartOf.Department  $\wedge$  isDepartmentType.Administration))

- A high-tech enterprise is an enterprise which has a research department.  
HighTechEnterprise = Enterprise  $\wedge$  hasDepartmentt.Research

- An industrial enterprise is an enterprise which has a production department and has at least 100 employees.

IndustrialEnterprise = Enterprise  $\wedge$  hasDepartment.Production

- A small enterprise is an enterprise which employs at most 20 employees.  
SmallEnterprise = Enterprise  $\wedge$  Employs.( $\leq$  20 Employee)

- A big enterprise is an enterprise which employs at least 80 employees  
BigEnterprise = Enterprise  $\wedge$  Employs.( $\geq$  80 Employee)

- A family-based enterprise is an enterprise with at most 4 employees.

FamilyBasedEnterprise = Enterprise  $\wedge$  Employs.( $\leq 4$  Employee)

- A top manager is someone who manages a big enterprise.

TopManager = Manages.BigEnterprise

- A manager is someone who manages a department.

Manager = Manages.Department

- A boss is someone who manages an office

Boss = Manages.Office

ABOX:

- Alcatel is an enterprise which has 2000 employees.

Enterprise(Alcatel)

NrEmployee(Alcatel, 2000) - adds anonymous employees to the dataset

- Alcatel has a research department RD1, an administration department AD1, and a HumanResources department HRD1; it has also a production department (whose name is unknown).

hasDepartmentType(Alcatel, Research)

hasDepartment(RD1, Research)

isPartOf(RD1, Alcatel)

hasDepartmentType(Alcatel, Administration)

hasDepartment(AD1, Administration)

isPartOf(AD1, Alcatel)

hasDepartmentType(Alcatel, HumanResources)

hasDepartment(HRD1, HumanResources)

isPartOf(HRD1, Alcatel)

hasDepartmentType(Alcatel, Production)

isPartOf(Production, Alcatel) - given the name production

- OFF1 and OFF2 are offices and are part of RD1.

Office(OFF1)

Office(OFF2)

isPartOf(OFF1, RD1)



isPartOf(OFF2, RD2)

- OFF3 and OFF4 are offices and are part of AD1.  
Office(OFF3)  
Office(OFF4)  
isPartOf(OFF3, AD1)  
isPartOf(OFF4, AD1)
- Joe and Anne are employed by OFF3.  
isEmployed(Joe, OFF3)  
isEmployed(Anne, OFF3)
- Jim manages the department AD3.  
Manages(Jim, AD3)
- Bob manages OFF3.  
Manages(Bob, OFF3)
- Jim manages Alcatel.  
Manages(Jim, Alcatel)
- SmithBrothers is a family-based enterprise.  
FamilyBasedEnterprise(SmithBrothers)
- Frank, Lea, Dave, Kate, Dino are employed by SmithBrothers  
isEmployed(Frank, SmithBrothers)  
isEmployed(Lea, SmithBrothers)  
isEmployed(Dave, SmithBrothers)  
isEmployed(Kate, SmithBrothers)  
isEmployed(Dino, SmithBrothers)

3. What conclusions can be derived from this TBOX+ABOX? Is it consistent?

From the TBOX and the facts could be derived:

- Alcatel is an enterprise which has 2000 employees.
  - Alcatel is a **BigEnterprise** because is an **Enterprise** and has more than **80 employees**
- Alcatel has a research department RD1, an administration department AD1, and a HumanResources department HRD1; it has also a production department (whose name is unknown).

- Given that the type of enterprises is not a disjoint set Alcatel can be more type of enterprises: Alcatel has a **Production Department** and given the previous facts (it has more than 800 employees) → Alcatel is an **IndustrialEnterprise**
- OFF1 and OFF2 are offices and are part of RD1.
  - No information is inferred and added to the Knowledge Base
- OFF3 and OFF4 are offices and are part of AD1.
  - No information is inferred and added to the Knowledge Base
- Joe and Anne are employed by OFF3.
  - From this can be deducted that Joe and Anne are **Employees** and are **AdministrativeEmployees**
- Jim manages the department AD3.
  - Jim is an **Employee** since he manages AD3.
  - Jim is also a **Manager**
  - **AD3 is a department**
- Bob manages OFF3.
  - Bob is an **Employee** since he manages OFF3.
  - Bob is also a **Boss**
  - Bob is also an **AdministrativeEmployee** is OFF3 **isPartOf** AD1 (administration department)
- Jim manages Alcatel
  - Jim is also a **TopManager** since Alcatel is a **BigEnterprise**
- SmithBrothers is a family-based enterprise.
  - The enterprise **SmithBrothers** should have a maximum of 4 members
- Frank, Lea, Dave, Kate, Dino are employed by SmithBrothers.
  - Frank, Lea, Dave, Kate and Dino are **Employees**
  - Found an inconsistency since there are hired 5 members in a **FamilyBasedEnterprise**

[0] <https://github.com/murarugeorgec/Enterprise>