SEMANTIC WEB
Unit-IV

Syllabus

Taxonomies and Ontologies: Overview of Taxonomies, Defining the Ontology Spectrum, Topic Maps, Overview of Ontologies, Syntax, Structure, Semantics, and Pragmatics, Expressing Ontologies Logically, Knowledge Representation.

Overview of Taxonomies

Here is the information technology definition for a taxonomy:

The classification of information entities in the form of a hierarchy, according to the presumed relationships of the real-world entities that they represent.

A taxonomy is usually depicted with the root of the taxonomy on top, as in Figure 7.1. Each node of the taxonomy—including the root—is an information entity that stands for a real-world entity. Each link between nodes represents a special relation called the *is subclassification of* relation (if the link's arrow is pointing up toward the parent node) or *is superclassification of* (if the link's arrow is pointing down at the child node). Sometimes this special relation is defined more strictly to be *is subclass of* or *is superclass of*, where it is understood to mean that the information entities (which, remember, stand for the real world entities) are classes of objects.

In Figure 7.1, examples include the class Person, its subclasses of Employee and Manager, and its superclass of Agent (a legal entity, which can also include an Organization, as shown in the figure).

As you go up the taxonomy toward the root at the top, the entities become more general. As you go down the taxonomy toward the leaves at the bottom, the entities become more specialized. Agent, for example, is more general than Person, which in turn is more general than Employee. This kind of classification system is sometimes called a *generalization/specialization* taxonomy.

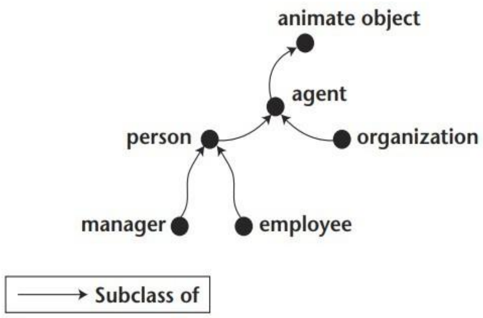

Figure 7.1 A simple taxonomy.

A taxonomy is a semantic hierarchy in which information entities are related by either the *subclassification of* relation or the *subclass of* relation. The former is semantically weaker than the latter, so we make a distinction between semantically weaker and semantically stronger taxonomies. Although taxonomies are fairly weak semantically to begin with—they don't have the complexity to express rich meaning—the stronger taxonomies try to use this notion of a distinguishing property. Each information entity is distinguished by a *distinguishing property* that makes it unique as a subclass of its parent entity (a synonym for *property* is *attribute* or *quality*).

Consider the distinction between mammal and reptile under their parent subphylum *Vertebrata* (in Figure 7.2, a dotted line between *Mammalia* and *Diapsida* shows that they are at the same level of representation, both being subclassifications of *Vertebrata*). Although both mammals and reptiles have four legs (common properties), mammals are warm-blooded and reptiles are cold- blooded. So *warm-bloodedness* can be considered at least one of the properties that distinguishes mammals and reptiles; there could be others. One other distinguishing property between mammals and reptiles is the property of *egg-laying*. Although there are exceptions (the Australian platypus, for example), mammals in general do not lay eggs, whereas reptiles do. (Reptiles also share this property with birds, fish, and most amphibians, but we will not elaborate that distinction here.)

- **Kingdom:** Animalia
  - **Phylum:** Chordata
    - **Subphylum:** Vertebrata
      - **Class:** Mammalia
        - **Subclass:** Theria
          - **Infraclass:** Eutheria
            - **Order:** Primates
              - **Suborder:** Anthropoidea
                - **Superfamily:** Hominoidea
                  - **Family:** Hominidae
                    - **Genus:** Homo
                      - **Species:** Sapiens
      - **Class:** Diapsida (Reptiles, Dinosaurs, Birds)

Figure 7.2 Linnaean classification of humans.

Why Use Taxonomies?

1. **Importance of Classifying Information**: it emphasizes the importance of organizing information by classifying it, similar to how libraries use the Dewey Decimal System. By giving structure and meaning to information, it becomes easier to find and navigate. This is crucial for both users searching for products or services and for businesses wanting their offerings to be discovered.

2. **Example of Internet Search:** When you search on the internet, you typically use keywords. If the information isn't classified or structured properly, it may not show up in search results, even if it's the best fit for what you're looking for. This means valuable products or services might go unnoticed, resulting in missed opportunities for both buyers and sellers.

3. **Purpose of Taxonomies:** Taxonomies, or classification systems, simplify the process of finding information, especially when you only have a vague idea of what you're looking for. They provide a framework for organizing diverse topics into categories, making it easier to navigate through a large amount of information.

4. **Comparison to Dewey Decimal System:** The Dewey Decimal System used in libraries is given as an example of a taxonomy. It divides subjects into broad categories, providing a structured way to organize books and help library visitors find what they're looking for.

In essence, the text stresses the importance of classifying information to make it easily discoverable and navigable, drawing parallels between internet searches and the organization systems used in libraries.

Topic Maps Concepts

The XTM standard15 identifies the key concepts of Topic Maps. The key concepts are *topic*, *association*, *occurrence*, *subject descriptor*, and *scope*. We describe these concepts in the following

Topic
Anything can be a topic—that is, any distinct subject of interest for which assertions can be made. A topic is a representation of the subject; according to the XTM standard, it acts as a resource that is a proxy for the subject.

A *subject* is the *what*—for instance, "Front Royal, Virginia" or "the Mars Lander" or "inventory control" or "agriculture"; a *topic* is an information representation of the *what*. So a topic represents the subject that is referred to. If the subject is “Front Royal,” then the topic would be Front Royal. Because subjects can be anything, topics can be anything. A topic is just a construct in Topic Maps, one of the essential building blocks. The way the subject of a topic is referred to is by having the topic point to a *resource* that expresses the subject. The resource either *constitutes* the subject (and so addresses the subject) or *indicates* the subject. In either case, the subject of the topic is represented by an *occurrence* of a resource, and it is the nature of that resource that determines the addressability of the subject. If the resource uses the *resourceRef* XTM construct, then it *constitutes* the subject and is addressable. If the resource uses the *subjectIndicatorRef* construct, then it *indicates* the subject and is not directly addressable. Web objects are addressable; non-Web objects are not directly addressable and so must be indicated (for example, all occurrences of the same topic are about the same subject, though they are distinct resources). A resource *occurrence* can also have a data value that is directly specified inline.

```
<topic id="Front Royal">
    <instanceOf><topicRef xlink:href="#city"/></instanceOf>
    <baseName>
      <baseNameString>Front Royal</baseNameString>
      <variant>
         <parameters><topicRef xlink:href="#display"/></parameters>
            <variantName>
               <resourceData>Gateway to Skyline Drive</resourceData>
            </variantName>
      </variant>
    </baseName>
    <occurrence>
      <instanceOf><topicRef xlink:href="#portal"/></instanceOf>
      <resourceRef
    xlink:href="http://www.ci.front-royal.va.us/"/>
    </occurrence>
  </topic>

  <topic id="Winchester">
    <instanceOf><topicRef xlink:href="#city"/></instanceOf>
    <baseName>
      <baseNameString>Winchester</baseNameString>
    </baseName>
    <occurrence>
      <instanceOf><topicRef xlink:href="#portal"/></instanceOf>
      <resourceRef xlink:href="http://www.ci.winchester.va.us/"/>
    </occurrence>
  </topic>
```

Listing 7.1 A Simple XTM topic map: Topics, occurrences.

In Listing 7.1, the topic map is enclosed by the <*topic*> and </*topic*> delimiters. The topic is identified by the *id="Front Royal"*. The topic is an *instance* of another topic, identified by the <*topicRef*> markup.

```
<instanceOf><topicRef xlink:href="#city"/></instanceOf>
```

In this case, Front Royal is a city, so the topic *Front Royal* is itself an instance of the topic reference *city*. Because the *resourceRef* construct is used, this example illustrates a topic that *constitutes* the subject, and the resource is addressable:

```
<occurrence>
        ...
        <resourceRef xlink:href="http://www.ci.front-royal.va.us/"/>
     </occurrence>
```

A topic is identified by a *name*. The primary way of identifying a topic map is to use the required *base name*. In the example, the base name of the topic is represented as:

```
<baseName>
            <baseNameString>Front Royal</baseNameString>
          ...
</baseName>
```

The <*basename*> and </*basename*> delimiters enclose this base name. The base name is meant to uniquely identify the topic (within a particular *scope*, which we will discuss later). In addition to the base name, however, a variant name, specifically, a *display name* and/or a *sort name*, can be used. In the example, a *display name* is represented, within the base name markup:

```
<variant>
        <parameters><topicRef xlink:href="#display"/></parameters>
           <variantName>
              <resourceData>Gateway to Skyline Drive</resourceData>
           </variantName>
</variant>
```

Each topic is implicitly an instance of a *topic type*—that is, the class of the topic, though the type may not be explicitly marked in any given topic map. If the topic type is not explicitly marked, then the topic is considered implicitly of type http://www.topicmaps.org/xtm/1.0/core.xtm#topic. A similar circumstance holds for typing *associations* and *occurrences*: If no type is specified, then an *association* or an *occurrence* is defined to be, respectively, of type http://www.topicmaps.org/xtm/1.0/core.xtm#association or http://www.topicmaps.org/xtm/1.0/core.xtm#occurrence.

Occurrence

As noted in the preceding text, an *occurrence* is a resource specifying some information about a topic. The resource is either addressable (using a URI) or has a data value specified inline. For the former, *resourceRef* is used. The example in Listing 7.1 illustrates this usage:

```
<occurrence>
        ...
        <resourceRef xlink:href="http://www.ci.front-royal.va.us/"/>
     </occurrence>
```

For the latter, the inline value, *resourceData*, is used (this is not part of Listing 7.1) for arbitrary character data:

```
<occurrence        ...
   <resourceData>Front Royal is on the Shenandoah River
   </resourceData>
</occurrence>
```

Note, however, that in Listing 7.1, the alternative use of *resourceData* is exemplified—not to specify an *occurrence*, but to specify a variant name:

```
<variantName>
        <resourceData>Gateway to Skyline Drive</resourceData>
</variantName>
```

Like topics, occurrences can also be of different types, specified by the *topicRef* markup. *Occurrences* are ways to characterize a topic. Because they can represent any information to be associated with a topic, they can also act as *attributes* of a topic, though XTM does not really distinguish attributes from other information, a distinction that is sometimes made in other schema or knowledge representation languages.

Association

An *association* is the relationship between (one or more) topics. Associations are delimited by <*association*> and </*association*>. In Listing 7.2, the association *located-in* is asserted to hold between two topic references: Front Royal (indicated by the URI that is the value of one *topicRef*) and Virginia (indicated by the URI that is the value of the other *topicRef*). The specification of the semantics of *located-in* is not explicitly represented but is assumed to be defined by or known to the creator of the topic map (and could remain implicit).

```
<association>
    <instanceOf><topicRef xlink:href="#located-in"/></instanceOf>
    <member>
      <roleSpec><topicRef xlink:href="#city"/></roleSpec>
      <topicRef xlink:href="#Front-Royal"/>
    </member>
    <member>
      <roleSpec><topicRef xlink:href="#state"/></roleSpec>
      <topicRef xlink:href="#Virginia"/>
    </member>
  </association>
```

Listing 7.2 Topic map associations.

As depicted in the preceding example, the association *located-in* is specified to be a (undirected) relationship between two members. A *member* is just a set of topics, in this case two topics identified as the URIs *#Front-Royal* and *#Virginia*, and demarcated by the *topicRef* constructs. This example also shows an important aspect of associations: The topics that are related by the association assume different *roles* in that association. The topic referenced as *#Front-Royal* is in the *#city* role, and the topic *#Virginia* is in the *#state* role of the *#located-in* association. An association *role* specifies how a particular topic acts as a member of an association, its manner of playing in that association. If there were a *uses* association between Sammy Sosa and a Rawlings 34-inch Pro

Model baseball bat, then Sammy would be in the *batter* role and the Rawlings would be in the *bat* role, as the following hypothetical portion of a topic map makes clear:

```
<association>
             <instanceOf><topicRef xlink:href="#uses"/></instanceOf>
             <member>
                <roleSpec><topicRef xlink:href="#batter"/></roleSpec>
                <topicRef xlink:href="#Sammy-Sosa"/>
             </member>
             <member>
                <roleSpec><topicRef xlink:href="#bat"/></roleSpec>
                <topicRef xlink:href="#Rawlings-34-inch-Pro-Model-
Baseball-Bat "/>
             </member>
     </association>
```

Subject Descriptor

We've looked at subjects in our discussion of topics. A *subject indicator* is just a way of indicating subjects. And topics are really the information representation of subjects. Typically (as we've seen), a subject is *indicated* by defining a *resource*. If two given topics in fact use the same resource, then their subjects (identified or indicated by those resources) are identical. For example, see Listing 7.3.

```
<topic id="Front Royal">
    <instanceOf><topicRef xlink:href="#city"/></instanceOf>
    <baseName>
      <baseNameString>Front Royal</baseNameString>
      <variant>
         <parameters><topicRef xlink:href="#display"/></parameters>
            <variantName>
               <resourceData>Gateway to Skyline Drive</resourceData>
            </variantName>
      </variant>
    </baseName>
    <occurrence>
      <instanceOf><topicRef xlink:href="#portal"/></instanceOf>
      <resourceRef
    xlink:href="http://www.ci.front-royal.va.us/"/>
    </occurrence>
  </topic>

  <topic id="Front Royal, Virginia">
    <instanceOf><topicRef xlink:href="#city"/></instanceOf>
```

Listing 7.3 Topic map subject indicators.

```
    <baseName>
      <baseNameString>Front Royal, Virginia</baseNameString>
    </baseName>
    <occurrence>
      <instanceOf><topicRef xlink:href="#portal"/></instanceOf>
      <resourceRef
    xlink:href="http://www.ci.front-royal.va.us/"/>
    </occurrence>
  </topic>
```

Listing 7.3 *(continued)*

In the listing, we'd like to say that both topics (*Front Royal* and *Front Royal, Virginia*) are really about the same subject. This judgment is confirmed, not by the near identity of the strings "Front Royal" and "Front Royal, Virginia" (whose string and concept similarity is apparent to a human being), but by the fact that both topics have the same resource or *subject indicator*, as represented by the common *occurrence* specification:

```
    <occurrence>
      <instanceOf><topicRef xlink:href="#portal"/></instanceOf>
      <resourceRef
    xlink:href="http://www.ci.front-royal.va.us/"/>
    </occurrence>
```

The XTM standard also allows for a *published subject indicator* or, more simply, a published subject. A *published subject* is simply a subject that has general definition and usage and is identified by a specific published reference. In fact, the XTM standard states that there are default, mandatory published subjects, made mandatory by the requirements of the XTM standard itself. They include *topic, association, occurrence, class-instance relationship, class, instance, superclass-subclass relationship, superclass, subclass, suitability for sorting*, and *suitability for display*.[17]

Scope

Scope in Topic Maps is similar to the notion of namespace in other markup languages. *Scope* specifies the applicability or context of the topic, its occurrences and resources, and its associations. Subjects have a scope. The names of topics are unique within a scope. Resources specified within a particular topic have the same scope as that topic. That is why topic maps should be merged if they have the same *base name*; they indicate the same subject having the same scope.

Overview of Ontologies.

Ontologies are about vocabularies and their meanings, with explicit, expressive, and well-defined semantics—possibly machine-interpretable.

Ontology Example

Figure 8.1 shows a simple human resources ontology created in the ontology management tool called Protégé (http://protégé.stanford.edu). You'll notice that there are classes such as Person, Organization, and Employee. In an ontology, these are really called *concepts*, because it is intended that they correspond to the mental concepts that human beings have when they understand a particular body of knowledge or subject matter area or domain (these phrases are all used interchangeably; they are intended to be synonymous), such as the human resources domain.

These concepts and the relationships between them are usually implemented as classes, relations, properties, attributes, and values (of the properties/attributes). So what Figure 8.1 depicts primarily are concepts of the important entities of the domain, which are implemented as classes. Examples are Person, Organization, and Employee. Also depicted are the relations between these entity-focused concepts, such as employee_of, managed_by, and manages. Finally, properties or attributes are depicted. Examples include address, name, birthdate, and ssn under the Person class. These properties or attributes have either explicit values or, more often, have value ranges. The value range for the property/attribute of employee_of, a property of the class Employee, for example, is the class Organization. By *range* we mean that the only possible values for any instances of the property employee_of defined for the class Employee must come from the class Organization.

Immediately we see that an ontology tries to capture the *meaning* (what we will call *semantics*) of a particular subject area or area of knowledge that corresponds to what a human being knows about that domain. An ontology also characterizes that meaning in terms of *concepts* and their relationships. Furthermore, an ontology is often represented as classes, relations, properties, attributes, and values.

Figure 8.1 is a graphical fragment of a simple ontology attempting to model the human resources domain (person, employee, organization), their subclasses (staff employee, management employee, company, group, division, and department), their properties, and the relationships among those concepts.

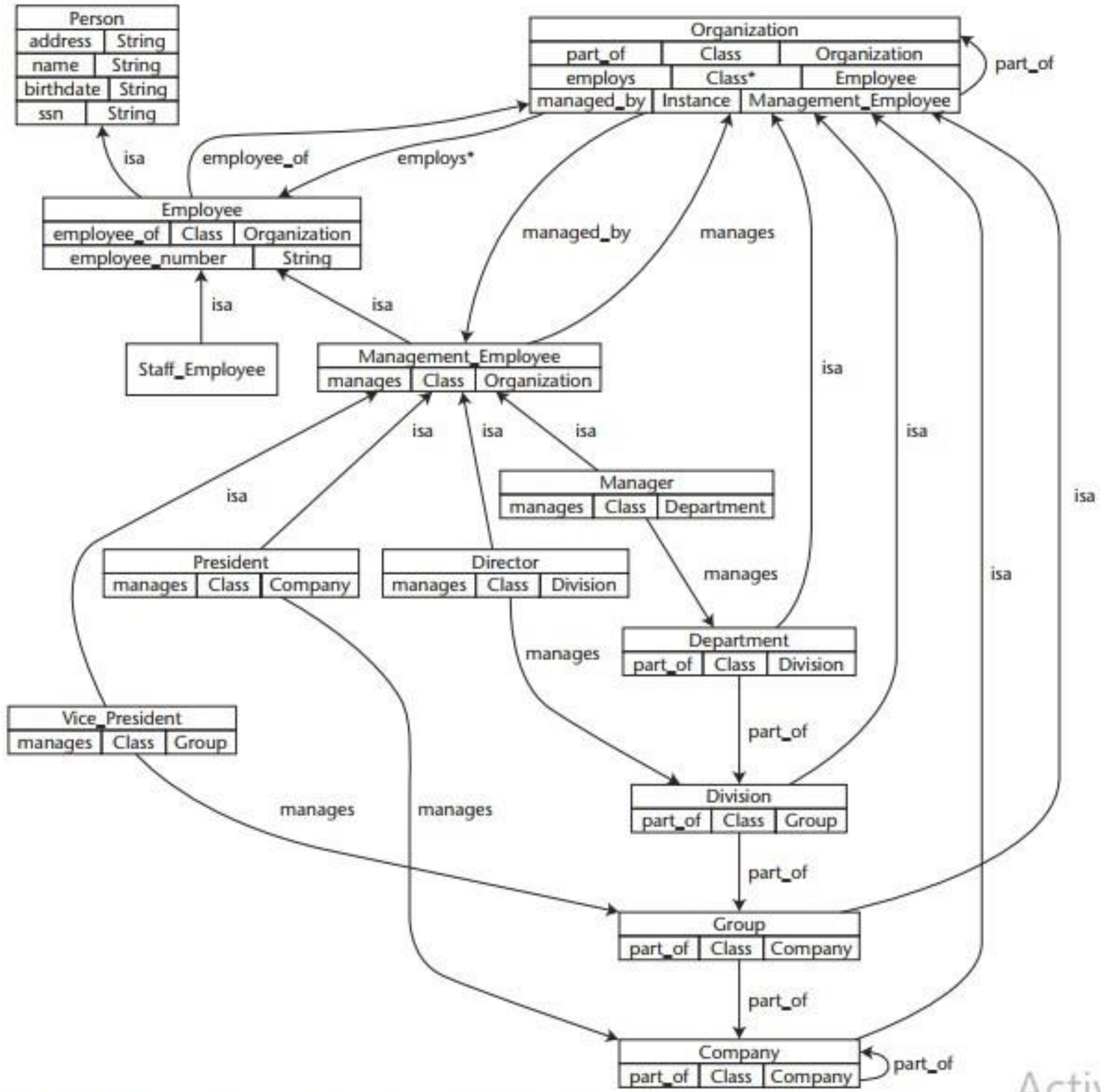

Figure 8.1 Graphical ontology example: Human resources.

Both underscore an important point: *There is no logical difference between a graphical and a textual endition of an ontology* (or any other model, for that matter). This fact is important, because a key point is that an ontology is represented in a *knowledge representation language* (such as a Semantic Web language like DF/S, DAML+OIL, OWL, or in an ontology language that predates the Semantic Web, such as Ontolingua/KIF/Common Logic, OKBC,CycL, or Prolog).