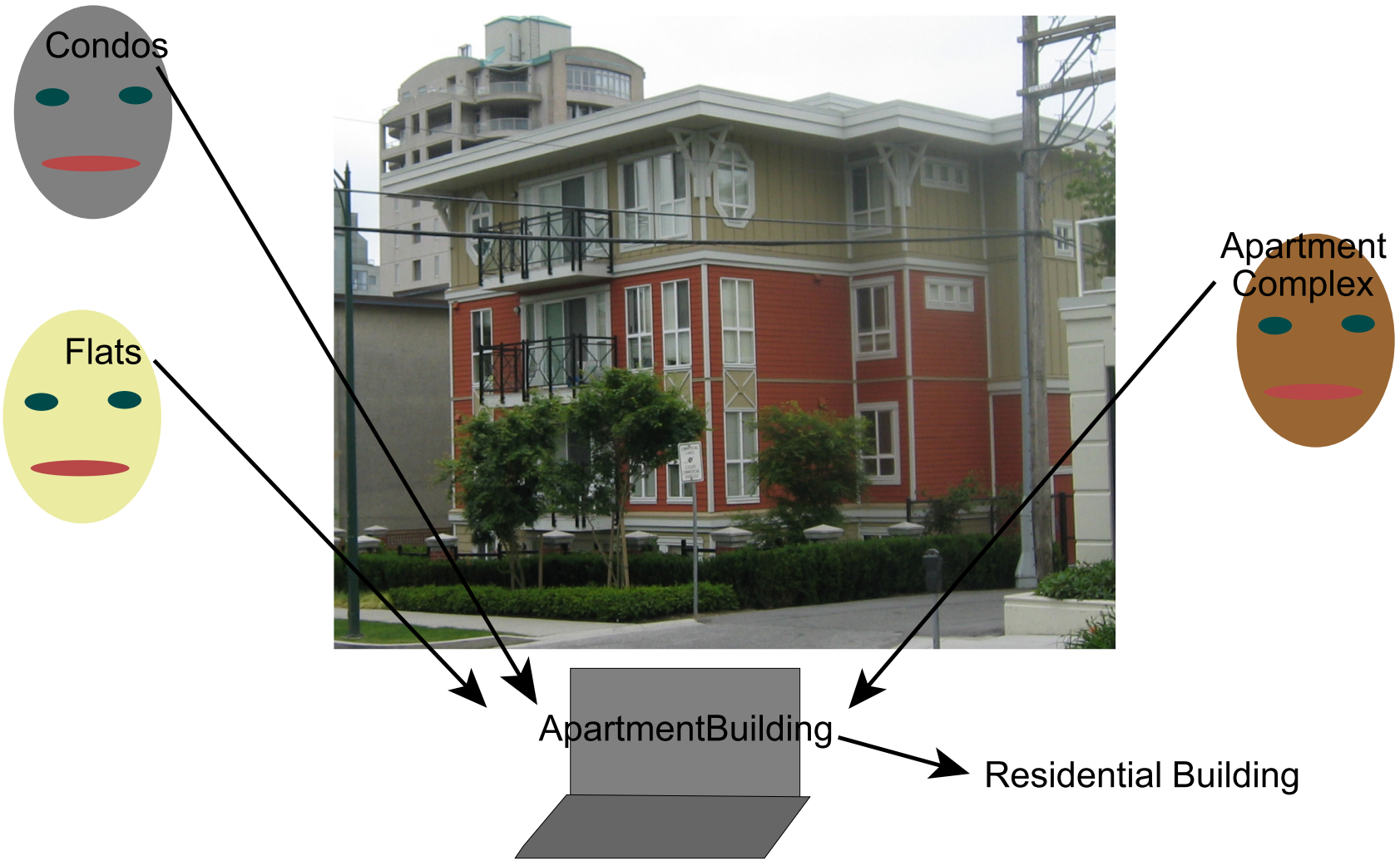
Knowledge Sharing

- If more than one person is building a knowledge base, they must be able to share the conceptualization.
- A conceptualization is a map from the problem domain into the representation. A conceptualization specifies:
  - What sorts of objects are being modeled
  - The vocabulary for specifying objects, relations and properties
  - The meaning or intention of the relations or properties
- An ontology is a specification of a conceptualization. An ontology specifies the meanings of the symbols in an information system.

Mapping from a conceptualization to a symbol



Semantic Web

- Ontologies are published on the web in machine readable form and are publicly readable.
- Builders of knowledge bases or web sites adhere to and refer to a published ontology:
  - the same symbol means the same thing across the various web sites that obey the ontology.
  - if someone wants to refer to some other object or relation, the ontology is expanded. The community needs to agree to the new terminology.

Challenges of building ontologies

- They can be huge: finding the appropriate terminology for a concept may be difficult.
- How one divides the world can depend on the application. Different ontologies describe the world in different ways.
- People can fundamentally disagree about the appropriate structure.
- Different knowledge bases can use different ontologies.
- To allow KBs based on different ontologies to inter-operate, there must be mapping between different ontologies.
- It has to be in user's interests to use an ontology.
- The computer doesn't understand the meaning of the symbols. The formalism can constrain the meaning, but can't define it.

Semantic Web Technologies

- XML provides generic syntax.
  $\langle tag \ldots /\rangle$ or
  $\langle tag \ldots \rangle \ldots \langle /tag \rangle$.
- URI is a name of an object (resource). This name can be shared. Often in the form of a URL to ensure uniqueness.
- RDF is a language of triples
- OWL the Web Ontology Language, defines some primitive properties that can be used to define terminology. (Doesn't define a syntax).

Main Components of an Ontology

- Individuals the objects in the world (not usually specified as part of the ontology)
- Classes sets of individuals
- Properties between individuals and their values

Individuals

- Individuals are things in the world that can be named. (Concrete, abstract, concepts, reified).
- Unique names assumption (UNA): different names refer to different individuals.
- The UNA is not an assumption you can universally make: "The Queen", "Elizabeth Windsor", etc.
- Without the unique names assumption, you can't count!
- In OWL you can specify:

  i_1 *SameIndividual* i_2.

  i_1 *DifferentIndividuals* i_3.

Classes

- A class is a set of individuals. E.g., house, building, officeBuilding
- One class can be a subclass of another

  house *subClassOf* *building*.

  officeBuilding *subClassOf* *building*.
- The most general class is *Thing*.
- Classes can be declared to be the same or to be disjoint:

  house *EquivalentClasses* *singleFamilyDwelling*.

  house *DisjointClasses* *officeBuilding*.
- Different classes are not necessarily disjoint.
  E.g., a building can be both a commercial building and a residential building.

Properties

- A property is between an individual and a value.
- A property has a domain and a range.

  livesIn domain person.

  livesIn range placeOfResidence.
- An *ObjectProperty* is a property whose range is an individual.
- A *DatatypeProperty* is one whose range isn't an object, e.g., is a number or string.
- There can also be a property hierarchies:

  livesIn subPropertyOf enclosure.

  principalResidence subPropertyOf livesIn.

Properties (Cont.)

- One property can be inverse of another

  livesIn *InverseProperties* *hasResident*.

- Properties can be declared to be transitive, symmetric, functional, or inverse-functional.
- You can also state the minimum and maximal cardinality of a property.

  principalResidence *minCardinality* 1.

  principalResidence *maxCardinality* 1.

Property and Class Restrictions

- You can define complex descriptions of classes in terms of restrictions of other classes and properties.
  E.g., A homeowner is a person who owns a house.

  $$homeOwner \quad subClassOf \quad person.$$

  $$homeOwner \quad subClassOf \quad SomeValuesFrom(owns, house).$$

Knowledge Sharing

- One ontology typically imports and builds on other ontologies.
- You need facilities for version control.
- Tools for mapping one ontology to another to allow inter-operation of different knowledge bases.
- The semantic web promises to allow you to find the right concept in a query if
  - the information adheres to some ontology
  - the query adhere to some ontology
  - these are the same ontology or there is a mapping between them.