Chapter 5. Describing Relationships and Structures

To appear in
The Discipline of Organizing, **2012**

Robert J. Glushko
Matthew Mayernik
Alberto Pepe

5.1 Introduction

A family is a collection of people affiliated by some connections with each other such as common ancestors or a common residence. The Simpson family includes a man named Homer and a woman named Marge, the married parents of three sibling children, a boy named Bart and two girls, Lisa and Maggie. Because this is an English-speaking family the boy describes his parents as his father and mother and his two siblings as his sisters. If instead this were a Spanish speaking family the boy would call his parents su padre and su madre and his sisters would be las hermanas. But if this were a Chinese family the boy would describe his sisters according to their ages relative to him; an older sister as zizi and his younger sister as meimei.[1]

Kinship relationships are ubiquitous and widely studied, and the names and significance of kinship relations like "is parent of" or "is sibling of" are familiar ones, making kinship a good starting point for understanding **relationships** in organizing systems.[2]

In a classic book called "Data and Reality" William Kent defines a **relationship** as "an association among several things, with that association having a particular significance."[3] The "things being associated," the components of the relationship, are people in kinship relationships but more generally can be any type of resource (Chapter 3), when we relate one resource instance to another. When we describe a resource (Chapter 4), the components of the relationship are a primary resource and a description resource. If we specify sets of relationships that go together, we are using these common relationships to define resource types or classes, which more generally are called categories (Chapter 6). We can then use resource types as one or both the components of a relationship when we want to further describe the resource type or to assert how two resource types go together to facilitate our interactions with them.

We begin with a more complete definition of "relationship" and introduce five perspectives for analyzing them: semantic, lexical, structural, architectural, and implementation. We then discuss each perspective, introducing the issues that each emphasizes and the specialized vocabulary needed to describe and analyze relationships from that point of view. We apply these perspectives and vocabulary to briefly analyze the most important types and applications of relationships in organizing systems.

5.2 Describing Relationships: An Overview

The concept of a "relationship" is pervasive in human societies in both informal and formal senses. Humans are inescapably related to generations of ancestors, and in most cases they also have social networks of friends, co-workers, and casual acquaintances to which they are related in various ways. We often hear that our access to information, money, jobs, and political power is all about "who you know," so we strive to "network" with other people to build relationships that might help us obtain them. In information systems, relationships between resources embody the organization that enables finding, selection, retrieval and other interactions.

A simple organizing system is like a relationship in that it defines an association among several things, but most organizing systems are based on many relationships that together enable the organizing system to satisfy some intentional purposes with individual resources or the collection as a whole. In the domain of information resources, common resources include web pages, journal articles, books, data sets, metadata records, and XML documents, among many others. Important relationships in the information domain that facilitate purposes like finding, identifying, and selecting resources include "is the author of", "is published by", "has publication date", "is derived from", "has subject keyword", "is linked to," and many others.

When we talk about relationships we specify both the resources that are associated along with a name or statement about the reason for the association. Just identifying the resources involved is not enough because several different relationships can exist among the same resources; the same person can be your brother, your employer, and your landlord. The order of the resources in the relationship usually matters; the person who is your employer gives a paycheck to you, not vice versa. Kent points out that when we describe a relationship we sometimes use whole phrases, such 'is-employed-by,'" if our language does not contain a single word that expresses the meaning of the relationship.

We can analyze relationships from several different perspectives.

- The **semantic** perspective is the most essential one; it characterizes the meaning of the association between resources.
- The **lexical** perspective focuses on how the conceptual description of a relationship is expressed using words in a specific language.
- The **structural** perspective analyzes the patterns of association, arrangement, proximity, or connection between resources.
- The **architectural** perspective emphasizes the number and abstraction level of the components of a relationship.
- The **implementation** perspective considers how the relationship is implemented in a particular notation and syntax and the manner in which relationships are arranged and stored in some technology environment.

5.3 The Semantic Perspective for Analyzing Relationships

In order to describe relationships among resources, we need to understand what the relations mean. This semantic perspective is the essence of relationships and explains why the resources are related, relying on information that is not directly available from perceiving the resources.[4] In our Simpson family example, we noted that Homer and Marge were related by marriage, and also by their relationship as parents of Bart, Lisa, and Maggie, and none of these relationships are directly perceivable.

A common way to specify a semantic relationship is by using a *subject-predicate-object* structure. For example, the relationship between two spouses could be expressed as follows:

Homer Simpson → is-married-to → Marge Simpson

This simple information structure, known as a triple, consists of a *subject* (Homer Simpson), an *object* (Marge Simpson), and a *predicate*, expressing the relationship existing between subject and object (is-married-to). Most relationships between resources can be expressed using a subject-predicate-object model.

However, we have not yet specified what the "is-married-to" relationship means. People can demonstrate their understanding of "is-married-to" by realizing that alternative and semantically equivalent expressions of the relationship between Homer and Marge might be

Marge Simpson → is-married-to → Homer Simpson
Homer Simpson → is-the-husband-of → Marge Simpson
Marge Simpson → is-the-wife-of → Homer Simpson

Going one step further, we could say that people understand the equivalence of these different expressions of the relationship because they have semantic and linguistic knowledge that relates some representation of "married," "husband," "wife," and other words. None of that knowledge is visible in the expressions of the relationships here, all of which specify concrete relationships about individuals and not abstract relationships between resource classes or concepts. We have simply pushed the problem of what it means to understand the expressions into the mind of the person doing the understanding.

We can be more rigorous and define the words used in these expressions so they are "in the world" rather than just "in the mind" of the person understanding them. We can write definitions:

- The marriage relationship is a consensual association of one person with another, most conventionally or traditionally one man and one woman, which is sanctioned by law and often by religious ceremonies
- A husband is a married man considered in relation to his wife.
- A wife is a married woman considered in relation to her husband.[5]

Definitions like these certainly help a person learn and make some sense of the relationship expressions involving Homer and Marge. However, the definitions are still not in a form that would enable someone to completely understand the Homer and Marge expressions because they rely on other undefined terms (consensual, law, etc.) and do not state the relationships among the concepts in the definitions. Furthermore, for a computer to understand the expressions, it needs a computer-processable representation of the relationships among words and meanings that makes every important semantic assumption and property precise and explicit. We will see what this takes starting in the next section.

5.3.1 Types of Semantic Relationships

In this discussion we will use "entity type," "class," "concept," and "resource type" as synonyms. "Entity type" and "class" are conventional terms in data modeling and database design, "concept" is the conventional term in computational or cognitive modeling, and "resource type" is the term we use when we discuss organizing systems. Similarly, we will

use "entity occurrence," "instance," and "resource instance" when we refer to one thing rather than to a class or type of them.

There are three broad categories of semantic relationships:

- **Inclusion:** one entity type contains or is comprised of other entity types; often expressed using "is-a," "is-a-type-of," "is part of," or "is-in" predicates
- **Attribution**: asserting or assigning values to properties; the predicate depends on the property: "is-the-author-of," "is-married-to," "is-employed-by," etc.
- **Possession**: asserting ownership or control of a resource; often expressed using a "has" predicate.

All of these are fundamental in organizing systems, both for describing and arranging resources themselves, and for describing the relationships among resources and resource descriptions.

5.3.1.1 Inclusion

There are three different types of inclusion relationships: class inclusion, meronymic inclusion, and topological inclusion. All three are commonly used in organizing systems.

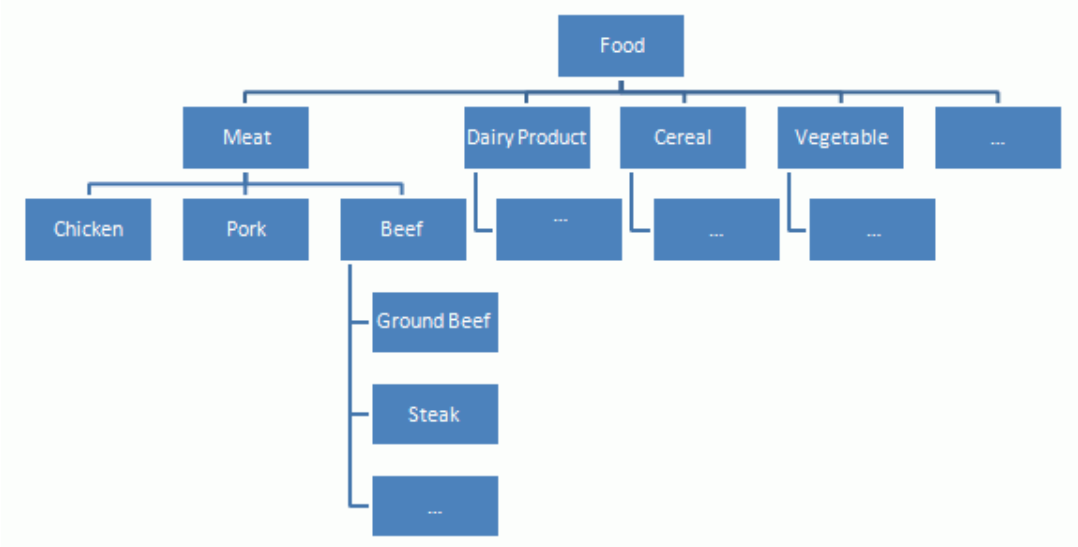
Class Inclusion is the fundamental and familiar "**is-a,**" "**is-a-type-of**," or "**subset**" relationship between two entity types or classes where one is contained in and thus more specific than the other more generic one.

Meat → is-a → Food

A set of interconnected class inclusion relationships creates a hierarchy, which is often called a **taxonomy**.

Dairy Product → is-a → Food
Cereal → is-a → Food
Vegetable → is-a → Food
Beef → is-a → Meat
Pork → is-a → Meat
Chicken → is-a → Meat
Ground Beef → is-a → Beef
Steak → is-a → Beef
…

Taxonomies are often depicted visually to make the class hierarchy easy to perceive.



Each level in a taxonomy subdivides the class above it into sub-classes, and each sub-class is further subdivided until the differences that remain among the members of each class no longer matter for the interactions the organizing system needs to support. We discuss the design of hierarchical organizing systems in Section 6.3, "Principles for Creating Categories."

All of the examples in the current section have expressed abstract relationships between classes, in contrast to the earlier concrete ones about Home and Marge, which expressed relationships between specific people. Homer and Marge are instances of classes like "married people," "husbands," and "wives." When we make an assertion that a particular instance is a member of class, we are **classifying** the instance. **Classification** is a class inclusion relationship between an instance and a class rather than between two classes (we discuss Classification in detail in Chapter 7).

Homer Simpson → is-a → Husband

This is just the lowest level of the class hierarchy in which Homer is located at the very bottom; he is also a man, a human being, and a living organism (in cartoon land, at least). You might now remember the bibliographic class inclusion hierarchy we discussed in Section 3.3.2; a specific physical **item** like your dog-eared copy of Macbeth is also a particular **manifestation** in some format or genre, and this **expression** is one of many for the abstract **work**.

Part-whole or **meronymic** inclusion is a second type of inclusion relationship. It is usually expressed using "is part of," "is partly," or with other similar predicate expressions. Winston, Chaffin, and Herrmann identified six distinct types of part-whole relationships:[6]

1. **Component- Object** is the relationship type when the part is a separate component that is arranged or assembled with other components to create a larger

resource. In Section 3.1.1.1, "Resources with Parts," we used as an example the component-object relationship between an engine and a car:

The Engine → is-part-of → the Car

The components of this type of part-whole relationship need not be physical objects; "Germany is part of the European Union" expresses a component-object relationship. What matters is that the component is identifiable on its own as an integral entity and that the components follow some kind of patterned organization or structure when they form the whole. Together the parts form a composition, and the parts collectively form the whole. If the whole ceases to exist neither do the parts, and vice versa.

2. **Member-Collection** is the part-whole relationship type where "is part of" means "is belongs to," a weaker kind of association than component-object because there is no assumption that the component has a specific role or function in the whole.

The Book → is-part-of → the Library

The members of the collection exist independently of the whole; if the whole ceases to exist the individual resource still exist.

3. **Portion-Mass** is the relationship type when all the parts are similar to each other and to the whole, unlike either of the previous types where engines are not tires or cars, and books are not like record albums or libraries.

The Slice → is-part-of → the Pie

4. **Stuff-Object** relationships are most often expressed using "is partly" or "is made of" and are distinguishable from component-object ones because the stuff cannot be separated from the object without altering its identity. The stuff is not a separate ingredient that is used to make the object; it is a constituent of it once it is made.

Wine → is-partly → Alcohol

5. **Place-Area** relationships exist between areas and specific places or locations within them. Like members of collections, places have no particular functional contribution to the whole.

The Everglades → are-part-of → Florida

6. **Feature-Activity** is a relationship type in which the components are stages, phases, or subactivities that take place over time. This relationship is similar to component-object in that the components in the whole are arranged according to a structure or pattern.

Overtime → is-part-of → a Football Game

A seventh type of part-whole relationship called **Phase-Activity** was proposed by Storey.[7] It is similar to feature-activity except that the phases do not make sense as standalone activities without the context provided by the activity as a whole.

Paying → is-part-of → Shopping

Topological or **Locative Inclusion** is a third type of inclusion relationship between a container, area, or temporal duration and what it surrounds or contains. It is most often expressed using "is-in" as the relationship. However, the included entity is not part of the including one, so this is not a part-whole relationship.

The Vatican City → is-in → Italy
The meeting → is-in → the afternoon

5.3.1.2 Attribution

Attribution relationships assert or assign values to properties. In Chapter 4 we used "attribute" to mean "an indivisible part of a resource description" and treated it as a synonym of "property." We now need to be more precise and carefully distinguish between the type of the attribute and the value that it has. The color of a car is an attribute of the car, and the value of that attribute might be green.

Some frameworks for semantic modeling define "attribute" very narrowly, restricting it to expressions with predicates with only one argument to assert properties of a single resource, distinguishing them from relationships between resources or resource types that require two arguments:

Martin the Gecko → is-small
Martin the Gecko → is-green[8]

However, it is always possible to express statements like these in ways that make them into relationships with two arguments:

Martin → has-size → small
Martin → has-skin-color → green

Dedre Gentner notes that this supposed distinction between one-predicate attributes and two-predicate relationships depends on context.[9] For example, small can be viewed as an attribute, **X → is-small,** or as a relationship between X and some standard or reference Y**, X -> is-smaller-than -> Y.**

Another somewhat tricky aspect of attribution relationships is that from a semantic perspective, many different ways of expressing the attribute value should be treated as equivalent.

Martin → has-size → 6 inches
Martin → has size → 152 mm

These two statements express the same idea, that Martin is small. However, many implementations of attribution relationships treat the attribute values literally. This means that unless we can process these two statements about Martin using another relationship that expresses the conversion of inches to mm, the two statements would be interpreted as saying different things about his size.

Finally, we note that we can express attribution relationships about other relationships, like the date a relationship was established. Homer and Marge Simpson's wedding anniversary is an attribute of their "is-married-to" relationship.

5.3.1.3 Possession

A third distinct category of semantic relationships is that of possession. The concept of possession is highly complex and inherently connected to societal norms and conventions about property and kinship. Possession relationships also imply duration or persistence, and are often difficult to distinguish from relationships based on habitual location or practice. Miller and Johnson-Laird illustrate the complex nature of possession relationships with this sentence, which expresses three different types of them:[10]

He owns an umbrella but she's borrowed it, though she doesn't have it with her.

Possession relationships can seem superficially like part-whole ones:

Bob → has → a car
A car → has → wheels

However, in the second of these relationships "has" is an elliptical form of "has as a part," expressing a component-object relationship rather that one of possession.

5.3.2 Properties of Semantic Relationships

Semantic relationships can have numerous special properties that help explain what they mean and especially how they relate to each other. In the following sections we briefly explain those that are most important in systems for organizing resources and resource descriptions.

5.3.2.1 Symmetry

In most relationships the order in which the subject and object arguments are expressed is central to the meaning of the relationship. If X has a relationship with Y, it is usually not the case that Y has the same relationship with X. For example, because "is-parent-of" is an **asymmetric** relationship, only the first of these relationships holds:

Homer Simpson → is-parent-of → Bart Simpson
Bart Simpson → is-parent-of → Homer Simpson

In contrast, some relationships are **symmetric** or **bi-directional**, and reversing the order of the arguments of the relationship predicate does not change the meaning. As we noted

earlier, these two statements are semantically equivalent because "is-married-to" is symmetric:

Homer Simpson → is-married-to → Marge Simpson
Marge Simpson → is-married-to → Homer Simpson

5.3.2.2 Transitivity

Transitivity is another property that can apply to semantic relationships. When a relationship is transitive, if X and Y have a relationship, and Y and Z have the same relationship, then X also has the relationship with Z. Any relationship based on ordering is transitive, which includes numerical, alphabetic, and chronological ones as well as those that imply qualitative or quantitative measurement. Because "is-older-than" is transitive,

Homer Simpson → is-taller-than → Bart Simpson
Bart Simpson → is-taller-than → Maggie Simpson

implies that

Homer Simpson → is-taller-than → Maggie Simpson

Inclusion relationships are inherently transitive, because just as "is-taller-than" is an assertion about relative physical size, "is-a-type of" and "is-part-of" are assertions about the relative sizes of abstract classes or categories.[11]

Transitive relationships enable inferences about class membership or properties, and allow organizing systems to be more efficient in how they represent them since transitivity enables implicit relationships to be made explicit only when they are needed.

5.3.2.3 Equivalence

Any relationship that is both symmetric and transitive is an **equivalence** relationship; "is-equal-to" is obviously an equivalence relationship because if A=B then B=A and if A=B and B=C, then A=C. Other relationships can be equivalent without meaning "exactly equal," as is the relationship of "is-congruent-to" for all triangles.

We often need to assert that a particular class or property has the same meaning as another class or property or that it is generally substitutable for it. We make this explicit with an equivalence relationship.

Wine (English language) → is-equivalent-to → Vin (French language)

5.3.2.4 Inverse

For asymmetric relationships, it is often useful to be explicit about the meaning of the relationship when the order of the arguments in the relationship is reversed. The resulting relationship is called the **inverse** or the **converse** of the first relationship. If an organizing system explicitly represents that:

Is-child-of → is-the-inverse-of → Is-parent-of

we can conclude that

Bart Simpson → is-child-of → Homer Simpson.

5.3.3 Ontologies

We now have described types and properties of semantic relationships in enough detail to return to the challenge we posed in the first part of Section 5.3: what information is required to fully understand relationships? This question has been posed and addressed for decades and we will not pretend to answer it to any extent here. However, we can sketch out some of the basic parts of the solution.

Let's begin by recalling that a taxonomy captures a system of class inclusion relationships in some domain. But as we have seen, there are a great many kinds of relationships that are not about class inclusion. All of these other types of relationships represent knowledge about the domain that is potentially needed to understand statements about it and to make sense when more than one domain of resources or activities comes together.

For example, in the food domain whose partial taxonomy appears in Section 5.3.1, we can assert relationships about properties of classes and instances, express equivalences about them, and otherwise enhance the representation of the food domain to create a complex network of relationships. In addition, the food domain intersects with food preparation, agriculture, commerce, and many other domains. We also need to express the relationships among these domains to fully understand any of them.

Hamburger → is-equivalent-to → Ground Beef
BigMac → is-a → Hamburger
A bun → is-part-of → the Hamburger
A bun → is-partly → flour
A Hamburger → is-prepared-by → Grilling
Grilling → is-a-type-of → Food Preparation
Temperate → is-a-measure-of → Grilling
Rare → is-a → Temperature
Well-done → is-a → Temperature
Meat → is-preserved-by → Freezing
Thawing → is-the-inverse-of → Freezing
…

In this simple example we see that class inclusion relationships form a kind of backbone or framework to which other kinds of relationships attach. We also see that there are vast numbers of potentially relevant assertions that together represent the knowledge that just about everyone knows about food and related domains. A network of relationships like these creates a resource that is called an **ontology.**[12]

<need a diagram here that shows the food taxonomy annotated with this additional information to create this ontology; note the graphical depiction is not the ontology… that's the computer processable representation>

There are numerous formats for expressing ontologies, but many of them have converged to or are based on OWL, the web ontology language developed by the W3C. OWL ontologies use a formal logic-based language that builds on RDF (Section 4.2.2.3) to define resource classes and assign properties to them in rigorous ways, arrange them in a class hierarchy, establish their equivalence, and specify the properties of relationships.[13]

Ontologies are essential parts in some organizing systems, especially information-intensive ones where the scope and scale of the resources require an extensive and controlled description vocabulary (See Section 4.3). The most extensive ontology ever created is Cyc, born in 1984 as an artificial intelligence research project. Three decades later, the latest version of the Cyc ontology contains several hundred thousand terms and millions of assertions that interrelate them.[14]

5.4 The Lexical Perspective for Analyzing Relationships

The semantic perspective for analyzing relationships is the fundamental one, but it is intrinsically tied to the lexical one because a relationship is always expressed using words in a specific language. For example, we understand the relationships among the concepts or classes of "food," "meat," and "beef" by using the words "food," "meat," and "beef" to identify progressively smaller classes of edible things in a class hierarchy.

However, the connection between concepts and words is not always this simple. In the Simpson family example with which we began this chapter, we noted in the contrast between "father" and "padre" that languages differ in the words they use to describe particular kinship relationships. Furthermore, we pointed out that cultures differ in which kinship relationships are conceptually distinct, so that languages like Chinese make distinctions about the relative ages of siblings that are not made in English.[15] This is not to suggest that an English speaker cannot notice the difference between his older and younger sisters, only that this distinction is not lexicalized – captured in a single word – as it is in Chinese. This "missing word" in English from the perspective of Chinese is called a **lexical gap**.[16]

Earlier in this book we discussed the naming of resources (Section 3.4.2) and the design of a vocabulary for resource description (Section 4.3.1.3), and we explained how increasing the scope and scale of an organizing system made it essential to be more systematic and precise in assigning names and descriptions. We need to be sure that the words we use to organize resources capture the similarities and differences between them well enough to support our interactions with them. After our discussion about semantic relationships in this chapter, we now have a clearer sense of what is required to bring like things together, keep different things separate, and to satisfy any other goals for the organizing system.

For example, if we are organizing cars, buses, bicycles, and sleds, all of which are vehicles, there is an important distinction between vehicles that are motorized and those that are powered by human effort. It might also be useful to distinguish vehicles with wheels from those that lack them. Not making these distinctions leaves an unbalanced or uneven organizing system for describing the semantics of the vehicle domain. However, only the "motorized" concept is lexicalized in English, which is why we needed to invent the "wheeled vehicle" term in the second case. [17]

Simply put, we need to use words effectively in organizing systems. To do that, we need to be careful about how we talk about the relationships among words and how words relate to concepts. There are two different contexts for those relationships. First, we need to discuss relationships among the meanings of words. Second, we need to discuss relationships among the form of words.

5.4.1 Relationships among Word Meanings

There are several different types of relationships of word meanings. Not surprisingly, in most cases they parallel the types of relationships among concepts that we described in Section 5.3.

5.4.1.1 Hyponymy and Hyperonymy

When words encode the semantic distinctions expressed by class inclusion, the word for the more specific class in this relationship is called the **hyponym,** while the word for the more general class to which it belongs is called the **hypernym**. George Miller suggested that an exemplary formula for defining a hyponym consists of its hypernym preceded by adjectives or followed by relative clauses that distinguish it from its **co-hyponyms**, mutually exclusive subtypes of the same hypernym.

hyponym = {adjective+} hypernym {distinguishing clause+}

For example, robin is a hyponym of bird, and could be defined as "a migratory bird that has a clear melodious song and a reddish breast with gray or black upper plumage." This definition does not describe every property of robins, but it is sufficient to differentiate robins from bluebirds or eagles.[18]

5.4.1.2 Metonomy

Part-whole or meronymic semantic relationships have lexical analogues in **metonomy**, when an entity is described by something that is contained in or otherwise part of it. A country's capital city or a building where its top leaders reside is often used as a metonym for the entire government; "The White House announced today…" Similarly, important concentrations of business activity are often metonyms for their entire industries: "Wall Street was bailed out again…"

5.4.1.3 Synonymy

Synonymy is the relationship between words that express the same semantic concept. The strictest definition is that synonyms "are words that can replace each other in some class of contexts with insignificant changes of the whole text's meaning."[19] This is an extremely

hard test to pass, except for acronyms or compound terms like "USA," "United States," and "United States of America" that are completely substitutable.

Most synonyms are not **absolute** synonyms like these, and instead are considered **propositiona**l ones. This means that they even though the words are not identical in meaning, they are equivalent enough in most contexts in that substituting one for the other will not change the truth value of the sentence that uses them. This weaker test lets us treat word pairs as synonyms even though their meanings differ in subtle ways. For example, if we know that Lisa Simpson plays the violin, because "violin" and "fiddle" are propositional synonyms, no one would disagree with an assertion that Lisa Simpson can play the fiddle.

An unordered set of synonyms is often called a **synset**, a term first used by the WordNet "semantic dictionary" project started in 1985 by George Miller and others at Princeton's Cognitive Science program.[20] Instead of using spelling as the primary organizing principle for words, WordNet uses their semantic properties and relationships to create a network that captures the idea that words and concepts are an inseparable system. Synsets are interconnected by both semantic relationships and lexical ones, enabling navigation in either space.[21]

5.4.1.4 Polysemy

We introduced the lexical relationship of **polysemy**, when a word has several different meanings or senses, in the context of problems with names (Section 3.4.2.2 Homonymy, Polysemy, and False Cognates). For example, the word "bank" can refer to any number of objects and activities: river bank, money bank, bank shots in basketball and billiards, an aircraft maneuver, to name a few.

Polysemy is represented in WordNet by including a word in multiple synsets. This enables WordNet to be an extremely useful resource for sense disambiguation in natural language processing research and applications. When a polysemous word is encountered, it and the words that are nearby in the text are looked up in WordNet. By following the lexical relationships in the synset hierarchy, a "synset distance" can be calculated. The smallest semantic distance between the words, which identifies their most semantically specific hypernym, can be used to identify the correct sense.[22]

5.4.1.5 Antonomy

Antonomy is the lexical relationship between two words that have opposite meanings. Antonymy is a very salient lexical relationship, and for adjectives it is even more powerful than synonymy. In word association tests, when the probe word is a familiar adjective, the most common response is its antonym; a probe of "good" elicits "bad," and vice versa. Like synonymy, antonomy is sometimes exact and sometimes more graded.[23]

Contrasting or binary antonyms are used in mutually exclusive contexts where one or the other word can be used, but never both. For example, "alive" and "dead" can never be used at the same time to describe the state of some entity, because the meaning of one excludes or contradicts the meaning of the other.

Other antonymic relationships between word pairs are less semantically sharp because they can sometimes appear in the same context as a result of the broader semantic scope of one of the words. "Large" and "small," or "old" and "young" generally suggest particular regions on size or age continua, but "how large is it?" or "how old is it?" can be asked about resources that are objectively small or young. [24]

5.4.2 Thesauri

The words that people naturally use when they describe resources reflect their unique experiences and perspectives, and this means that people often use different words for the same resource and the same words for different ones. Guiding people when they select description words from a controlled vocabulary is a partial solution to this vocabulary problem (Section 3.4.2.1) that becomes increasingly essential as the scope and scale of the organizing system grows. A **thesaurus** is a tool used by professionals when they describe resources that uses the lexical relationships we have introduced in this chapter to suggest which terms to use.

Thesauri have been created for many domains and subject areas. Some thesauri are very broad and contain words from many disciplines, like the Library of Congress Subject Headings used to classify any published content. Other commonly used thesauri are more focused, like the Art and Architecture Thesaurus developed by the Getty Trust and the Legislative Indexing Vocabulary developed by the US Library of Congress.[25]

We can return to our simple food taxonomy to illustrate how a thesaurus annotates vocabulary terms with lexical relationships. The class inclusion relationships of hyperonomy and hyponymy are usually encoded using BT ("broader term") and NT ("narrower term"):

Food BT Meat
Beef NT Meat

The BT and NT relationships in a thesaurus create a hierarchical system of words, but a thesaurus is more than a lexical taxonomy for some domain because it also encodes additional lexical relationships for the most important words. Many thesauri emphasize the cluster of relationships for these key words and de-emphasize the overall lexical hierarchy.

Because the purpose of a thesaurus is to reduce synonymy, it distinguishes among synonyms or near-synonyms by indicating one of them as a preferred term using UF ("used for"):

Food UF Sustenance, Nourishment

A thesaurus might employ USE as the inverse of the UF relationship to refer from a less preferred or variant term to a preferred one:

Victuals USE Food

Thesauri also use RT ("related term" or "see also") to indicate terms that are not synonyms but which often occur in similar contexts:

Food RT Cooking, Dining, Cuisine

5.4.3 Relationships among Word Forms

The relationships among word meanings are critically important, but whenever we create, combine, or compare resource descriptions we also need to pay attention to relationships between word forms. These relationships begin with the idea that all natural languages create words and word forms from smaller units. These basic building blocks are called **morphemes** and can express semantic concepts (when they are called **root** words) or abstract concepts like "pastness" or "plural"). The analysis of the ways by which languages combine morphemes is called **morphology**.[26]

Morphological analysis of a language is heavily used in text processing to create indexes for information retrieval; for example, **stemming** (discussed in more detail in Chapter 9) is morphological processing to remove prefixes and suffixes to leave the root form of words. Similarly, simple text processing applications like hyphenation and spelling correction solve word form problems using roots and rules because it is more scaleable and robust than solving them using word lists. Many misspellings of common words are obscure low frequency words, so adding them to a misspelling list would make it impossible to check spellings for the latter. In addition, because natural languages are generative and create new words all the time, a word list can never be complete.

5.4.2.1 Derivational Morphology

Derivational morphology deals with how words are created by combining morphemes. **Compounding**, putting two "free" morphemes together as in "batman" or "catwoman," is an extremely powerful mechanism. The meaning of some compounds is easy to understand when the first morpheme qualifies or restricts the meaning of the second, as in "doghouse" and "tollbooth."[27] However, many compounds take on new meanings that are not as literally derived from the meaning of their constituents, like "seahorse" and "batman."

Other types of derivations using "bound" morphemes follow more precise rules for combining them with "base" morphemes. The most common types of bound morphemes are prefixes and suffixes, which usually create a word of a different part-of-speech category when they are added. Familiar English prefixes include "a-," "ab-," "anti-," "co-," "de-," "pre-," and "un-." Among the most common English suffixes are "-able," "-ation," "-ify," "ing," "-ity," "-ize," "-ment," and "-ness." Compounding and adding prefixes or suffixes are simple mechanisms, but very complex words like "unimaginability" can be formed by using them in combination.

5.4.2.2 Inflectional Morphology

Inflectional mechanisms change the form of a word to represent tense, aspect, agreement, or other grammatical information. Unlike derivation, inflection never changes the part-of-

speech of the base morpheme. The inflectional morphology of English is relatively simple compared with other languages; for example, in Classical Greek each noun can have 11 word forms, each adjective 30, and every regular verb over 300.[28]

5.5 The Structural Perspective for Analyzing Relationships

The **structural** perspective analyzes the presence of association, arrangement, proximity, or connection between resources without primary concern for their meaning or origin of these relationships. We take a structural perspective when we define a family as "a collection of people" or when we say that a particular family like the Simpsons has five members. Sometimes all we know is that two resources are connected, as when we see a highlighted link pointing from one web page to another. At other times we might know more about the reasons for the relationships within a set of resources, but we still focus on their structure, essentially merging or blurring all of the reasons for the associations into a single generic notion that the resources are connected. We do this when we analyze communication or interaction patterns to determine the number of "degrees of separation" between any pair of resources.[29]

Many types of resources have internal structure in additional to their structural relationships with other resources. Of course, we have to remember (as we discussed in Section 3.3) that we often face arbitrary choices about the abstraction and granularity with which we describe the parts that make up a resource and whether some combination of resource should also be identified as a resource. This is not easy when you are analyzing the structure of a car with its thousands of parts, and it is ever harder with information resources where there are many more ways to define parts and wholes. However, an advantage for information resources is that their internal structural descriptions are usually highly "computable," something we consider in depth in Chapter 9.

5.5.1 Intentional, Implicit, and Explicit Structure

In the discipline of organizing we emphasize intentional structure created by people or by computational processes rather than accidental or naturally-occurring structures created by physical, geological, biological or genetic processes. We acknowledged in Section 1.2.3 that there is information in the piles of debris left after a tornado or tsunami and in the strata of the Grand Canyon. Similarly, we can perceive a pattern of stars and name it Orion or the Big Dipper, but this structural organization only exists from our galactic point of view; the stars that make up these constellations are at significantly different distances from Earth. These structural patterns might be of interest to meteorologists, geologists, astronomers or others but because they were not created by an identifiable agent following one or more organizing principles, they are not our primary focus.

Some organizing principles impose very little structure. For a small collection of resources, co-locating them or arranging them near each other might be sufficient organization. We can impose two- or three-dimensional coordinate systems on this implicit structure and explicitly describe the location of a resource as precisely as we want, but we more naturally describe the structure of resource locations in relative terms. In English we have many ways to describe the structural relationship of one resource to another: "in," "on," "under,"

"behind," "above," "below," "near," "to the right of," "to the left of," "next to," and so on. Sometimes several resources are arranged or appear to be arranged in a sequence or order and we can use positional descriptions of structure: a late 1990s TV show described the planet Earth as "the third rock from the Sun."[30]

We pay most attention to intentional structures that are explicitly represented within and between resources because they embody the design or authoring choices about how much implicit or latent structure will be made explicit. For very large collections of resources whose scope and scale defies structural analysis by people, structures that can be reliably extracted by algorithms are the most important.

5.5.2 Structural Relationships within a Resource

We almost always think of human and other animate resources as unitary entities. Likewise, many physical resources like paintings, sculptures, and manufactured goods have a material integrity that makes us usually consider them as indivisible. For an information resource, however, it is almost always the case that it has or might have had some internal structure.

When an author writes a document, he or she gives it some internal organization with its title, section headings, typographic conventions, page numbers, and other mechanisms that identify its parts and their significance or relationship to each other.

In data-intensive or transactional domains, document instances tend to be homogeneous because they are produced by or for automated processes, and their information components will appear predictably in the same structural relationships with each other. These structures typically form a hierarchy expressed in an XML schema or word processing style template. XML documents describe their component parts using content-oriented elements like <ITEM>, <NAME>, and <ADDRESS> that themselves are often aggregate structures or containers for more granular elements. The structures of resources maintained in databases are typically less hierarchical, but the structures are precisely captured in database schemas.

The internal parts of XML documents can be described and found using the XPath language, which defines the structures and patterns used by XML forms, queries, and transformations. The key idea used by XPath is that the structure of XML documents is a tree of information items called nodes, whose locations are described in terms of the relationships between nodes. The relationships built into XPath, which it calls axes, include self, child, parent, following, and preceding, making it very easy to specify a structure-based query like "find all sections in chapters 1-5 that have at least two levels of subsections."[31]

In more qualitative, less information-intensive and more experience-intensive domains, we move toward the narrative end of the document type spectrum, and document instances become more heterogeneous because they are produced by and for people. The information conveyed in the documents is conceptual or thematic rather than transactional, and the structural relationships between document parts are much weaker.

Instead of precise structure and content rules,, there is usually just a shallow hierarchy marked up with Word processing or HTML tags like <HEAD>, <H1>, <H2>, and <LIST>.

The internal structural hierarchy in a resource is often extracted and made into a separate and familiar description resource called the "table of contents" to support finding and navigation interactions with the primary resource. Some tables of contents are created as a static structural description, but others are dynamically generated from the internal structures whenever the resource is accessed. In additional, other types of entry points can be generated from the names or descriptions of content components, like selectable lists of tables, figures, maps, or code examples.

Identifying the components and their structural relationships in documents is easier when they follow consistent rules for structure (e.g., every non-text component must have a title and caption) and presentation (e.g., hypertext links in web pages are underlined and change cursor shapes when they are "moused over") that reinforce the distinctions between types of information components. Structural and presentation features can sometimes be ordered on some dimension (e.g., type size, line width, amount of white space) and used in a correlated manner to indicate the importance of a content component.[32]

Many indexing algorithms treat documents as "bags of words" to compute statistics about the frequency and distribution of the words they contain while ignoring all semantics and structure. In Chapter 9 we contrast this approach with algorithms that use internal structural descriptions to retrieve more specific parts of documents.

5.5.3 Structural Relationships between Resources

Many types of resources have structural relationships that interconnect them. Web pages are almost always linked to other pages. Sometimes the links among a set of pages remain mostly within those pages, as they are in an e-commerce catalog site. More often, however, links connect to pages in other sites, creating a link network that cuts across and obscures the boundaries between sites.

The links between documents can be analyzed to infer connections between the authors of the documents. Using the pattern of links between documents to understand the structure of knowledge and of the intellectual community that creates it is not a new idea, but it has been energized as more of the information we exchange with other people is on the web or otherwise in digital formats. An important function in Google's search engine is the **page rank** algorithm that calculates the relevance of a page in part using the number of links that point to it while giving greater weight to pages that are themselves linked to often.[33]

Web-based social networks enable people to express their connections with other people directly, bypassing the need to infer the connections from links in documents or other communications.

5.5.3.1 Hypertext Links

The concept of hypertext links as read-only or follow-only structures that connect one document to another is usually attributed to Vannevar Bush in his seminal essay titled "As We May Think." Bush called it "associative indexing," defined as "a provision whereby any item may be caused at will to select immediately and automatically another."[34] The "item" connected in this way was for Bush most often a book or a scientific article. However, the **anchor** and **destination** of a hypertext link can be a resource of any granularity, ranging from a single point or character, a paragraph, a document, or any part of the resource to which the ends of link are connected.

The anchor and destination of a link are its structural specification, but we often need to consider links from other perspectives. See the SIDEBAR "Perspectives on Hypertext Links."

PERSPECTIVES ON HYPERTEXT LINKS

Many hypertext links are purely structural because there is no explicit representation of the reason for the relationship. When one exists, this semantic property of the link is called the **link type.**[35] Sometimes the relationship is meaningful in both directions, implying an inverse relationship for the (sometimes implicit) back link from the destination to the anchor.

A lexical perspective on hypertext links concerns the words that are used to signal the presence of a link or to encode its type. In web contexts the words in which the structural link is embedded are called the **anchor text**. More generally, rhetorical structure theory analyzes how different conventions or signals in texts indicate relationships between texts or parts of them, like the subtle differences in polarity among "see," "see also," and "but see" as citation signals. [36]

An architectural perspective on links considers whether links are **one-way** or **bi-directional**; when a bi-directional link is created between an anchor and a destination, it is as though a one-way link that can be followed in the opposite direction is automatically created. Two one-way links serve the same purpose, but the return link is not automatically established when the first one is created. A second architectural consideration is whether links are **binary**, connecting one anchor to one destination, or **n-ary**, connecting one anchor to multiple types of destinations. [37] (See Section 5.6).

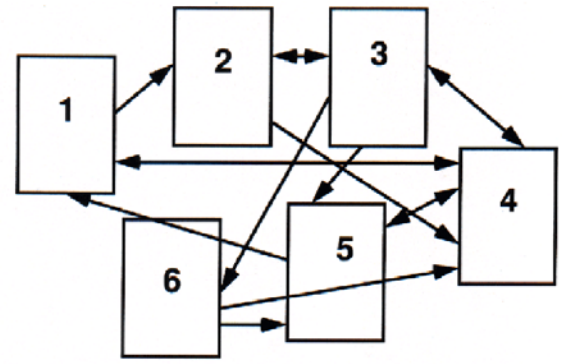
A "front end" or "surface" implementation perspective on hypertext links concerns how the presence of the link is indicated in a user interface; this is called the **link marker**; underlining or coloring of clickable text are conventional markers for web links.[38] A "back end" implementation issue is whether links are contained or embedded in the resources they link or whether they are stored separately in a **link base**.[39] (See Section 5.7).

Ted Nelson renamed associative indexing as **hypertex**t about twenty years later, expanding the idea to make it a writing style as well as a reading style. Nelson urged writers to use hypertext to create non-sequential narratives that gave choices to readers.[40]

The resources connected by hypertext links are not limited to text or documents. Selecting a hypertext link can reveal or invoke a connected resource that might be a picture, video, or interactive application.[41] More generally, hypertext links are frequently used as state transition controls in distributed collections of web-based resources.[42]

5.5.3.2 Analyzing Link Structures

We can portray a set of links between resources graphically as a pattern of boxes and links. Because a link connection from one resource to another need not imply a link in the opposite direction, we distinguish one-way links from explicitly bi-directional ones.



For a small network of links, a diagram like this one makes it easy to see that some resources have more incoming or outgoing links than other resources. But for most purposes we leave the analysis of link structures to computer programs, and there it is much better to represent the link structures more abstractly in matrix form. In this matrix the resource identifiers on the row and column heads represent the source and destination of the link. This is a full matrix because it is not symmetric; a link from resource 1 to resource 2 does not imply one from 2 to 1.

	1	2	3	4	5	6
1		x				
2			x	x		
3		x		x	x	x
4	x		x		x	
5	x			x		
6				x	x	

This representation models the network as a directed graph in which the resources are the vertices and the relationships are the edges that connect them. We now can apply graph

algorithms to determine many useful properties. A very important property is **reachability,** the "can you get there from here" property, which is determined by calculating the **transitive closure** of the matrix.[43] Other useful properties include the average number of incoming or outgoing links, the average distance between any two resources, and the shortest path between them,

5.5.2.3 Bibliometrics, Shepardizing, and Social Network Analysis

Information scientists began studying the structure of scientific citation, now called **bibliometrics**, nearly a century ago to identify influential scientists and publications. This analysis can identify **invisible colleges** of scientists who rely on each other's research, and recognize the emergence of new scientific disciplines or research areas.[44]

The expression of citation relationships between documents is especially nuanced in legal contexts, where the use of legal cases as precedents makes it essential to distinguish precisely where a new ruling lies on the relational continuum between "Following" and "Overruling" with respect to a case it cites. The analysis of legal citations to determine whether a cited case is still good law is called **Shepardizing** because lists of cases annotated in this way were first published in the late 1800s by Frank Shepard, a salesman for a legal publishing company.[45]

Facebook's multi-billion dollar valuation after its 2012 initial public offering is based on its ability to exploit the structure of a person's social network to personalize advertisements for people and their "friends" to whom they are connected. Many computer science researchers are working to determine the important characteristics of people and relationships that best identify the people whose activities or messages influence others to spend money.[46]

5.6 The Architectural Perspective for Analyzing Relationships

The **architectural** perspective emphasizes the number and abstraction level of the components of a relationship, which together characterize the complexity of the relationship. We will briefly consider three architectural issues: degree (or arity), cardinality, and directionality.

5.6.1 Degree

The **degree** or **arity** of a relationship is the number of entity types or categories of resources in the relationship. This is usually, though not always, the same as the number of arguments in the relationship expression.

Homer Simpson → is-married-to → Marge Simpson

Is a relationship of degree 2, a **binary** relationship between two entity types, because the "is-married-to" relationship as we first defined it requires one of the arguments to be of entity type "husband" and one of them to be of type "wife."

Now suppose we change the definition of marriage to make it less restrictive by allowing its arguments to be any instance of the entity type "person." The relationship expression looks exactly the same, but its degree is now **unary** because only 1 entity type is needed to instantiate the two arguments,

Some relationships are best expressed as **ternary** ones that involve three different entity types. An example that appears in numerous data modeling books is one like this:

Supplier → provides → Part → assembled-in → Product

It is always possible to represent ternary relationships as a set of binary ones by creating a new entity type that relates to each of the others in turn. This new entity type is called a dummy in modeling practice.

Supplier → provides → DUMMY
Part → provided-for → DUMMY
DUMMY → assembled-in → Product

This transformation from a sensible ternary relationship to three binary ones involving a DUMMY entity type undoubtedly seems strange, but it enables all relationships to be binary while still preserving the meaning of the original ternary one. Making all relationships binary makes it easier to store relationships and combine them to discover new ones

5.6.2 Cardinality

The **cardinality** of a relationship is the number of instances that can be associated with each entity type in a relationship. At first glance this might seem to be degree by another name, but it is not,

Cardinality is easiest to explain for binary relationships. If we return to Homer and Marge, the binary relationship that expresses that they a married husband and wife is a **one-to-one** relationship because a husband can only have one wife and a wife can only have one husband (at a time, in monogamous societies like the one that the Simpsons live in),

In contrast, the "is-parent-of" relationship is one-to-many, because the meaning of being a parent makes it correct to say that

Homer Simpson → is-parent-of → Bart AND Lisa AND Maggie

As we did with the ternary relationship in Section 5.6.1, we can transform this more complex relationship architecture to a set of simpler ones by restricting expressions about being a parent to the one-to-one cardinality.

Homer Simpson → is-parent-of → Bart
Homer Simpson → is-parent-of → Lisa
Homer Simpson → is-parent-of → Maggie

As before, there are implications of this transformation. The one-to-many expression brings all three of Homer's children together as arguments in the same relational expression, making it immediately obvious that they are siblings. We can infer this from the set of separate and redundant one-to-one expressions, but that involves additional logical processing to reconstruct what was explicit in the one-to-many case.

5.6.3 Directionality

The **directionality** of a relationship defines the order in which the arguments of the relationship are connected. A **one-way** or **uni-directional** relationship can be followed in only one direction, whereas a **bi-directiona**l one can be followed in both directions.

All symmetric relationships (Section 5.3.2.1) are bi-directional, but not all bi-directional relationships are symmetric. A relationship between a manager and an employee that he manages is "employs," a different meaning than the "is-employed-by" relationship in the opposite direction. As in this example, the relationship is often lexicalized in only one direction.

5.7 The Implementation Perspective for Analyzing Relationships

A final perspective on relationships is **implementation**, which considers how a relationship is realized or encoded in a technology context. The implementation perspective contrasts strongly with the conceptual, structural, and architectural perspectives, which emphasize the meaning and abstract structure of relationships. The implementation perspective is a superset of the lexical perspective, because the choice of the language in which to express a relationship is an implementation decision. However, most people think of implementation as all of the decisions about technological form or context rather than just about the choice of words.

In this book we take a narrower perspective on the implementation of relationships and relationship descriptions because we want to focus on the more fundamental issues and challenges that apply to all organizing systems, and not just on information-intensive ones that rely extensively on technology. Even with this reduced scope, there are some critical implementation concerns about the notation, syntax, and deployment of the relationships and other descriptions about resources. We briefly introduce some of these issues here and then discuss them in detail in Chapter 8, "The Form of Resource Description."

The syntax and grammar of a language consists of the rules that determine which combinations of its words are allowed and are thus **grammatical** or **well-formed.** Natural languages differ immensely in how they arrange nouns, verbs, adjectives, and other parts of speech to create sentences. Conformance to these rules makes the sentence syntactically compliant but does not mean that an expression is semantically comprehensible; the classic example is Chomsky's anomalous sentence:

Colorless green ideas sleep furiously

This sentence is nonsense, but anyone who knows the English language can recognize that it follows its syntactic rules, as opposed to this sentence, which breaks them:

Ideas colorless sleep furiously green[47]

The most basic requirement for an implementation syntax is that it can represent all the expressions that it needs to express. For the examples in this chapter we used an informal combination of English words and symbols (arrows and parentheses) that you could understand easily, but simple language is incapable of expressing most of what we readily say in English.

A second consideration is that the implementation can be understood and used by its intended users. We can usually express a relationship in different languages while preserving its meaning, just as we can usually implement the same computing functionality in different programming languages. From a semantic perspective these three expressions are equivalent:

My name is Homer Simpson
Mon nom est Homer Simpson
Mein Name ist Homer Simpson

However, whether these expressions are equivalent for someone reading them depends on which languages they understand.

An analogous situation occurs with the implementation of web pages. HTML was invented as a language for encoding how web pages look in a browser, and most of the tags in HTML represent formatting and simple structure. This makes the idea of using HTML so conceptually straightforward that school children can understand HTML and create web pages. However, the "web for eyes" implemented using HTML is of little use to computers, which want to treat web content as product catalogs, orders, invoices, payments, and other business information that can be analyzed and processed. This "web for computers" is best implemented using XML.

5.8 Relationships in Organizing Systems

In the previous sections as we surveyed the five perspectives on analyzing relationships we mentioned numerous examples where relationships had important roles in organizing systems. In this final section we examine in a bit more detail three contexts for organizing systems where relationships are especially fundamental; the Semantic Web and Linked Data, bibliographic organizing systems, and situations involving system integration and interoperability,

5.8.1 The Semantic Web and Linked Data

In a classic 2001 paper, Tim Berners-Lee laid out a vision of a Semantic Web in which all information could be shared and processed by automated tools as well as by people.[48] The essential technologies for making the web more semantic and relationships among web resources more explicit are XML, RDF (Section 4.2.2.3), and OWL (Section 5.3.3), Many

tools have been developed to support more semantic encoding, but most still require substantial expertise in semantic technologies and web standards.[49] More likely to succeed are applications that aim lower, not trying to encode all the latent semantics in a document or web page. For example, some wiki and blogging tools contain templates for semantic annotation, and Wikipedia has thousands of templates and "infoboxes" to encourage the creation of information in content-encoded formats.

The "Linked Data" movement is an extension of the Semantic Web idea to reframe the basic principles of the web's architecture in more semantic terms. Instead of the limited role of links as simple untyped relationships between HTML documents, links between resources described by RDF can serve as the bridges between islands of semantic data, creating a Linked Data network or cloud. [50]

5.8.2 Bibliographic Organizing Systems

Much of our thinking about relationships in organizing systems for information comes from the domain of bibliographic cataloguing of library resources and the related areas of classification systems and descriptive thesauri. Bibliographic relationships provide an important means to build structure into library catalogs.[51]

Bibliographic relationships are common among library resources. Smiraglia and Leazer found that approximately 30% of the works in the Online Computer Library Center's (OCLC) WorldCat union catalog have associated derivative works. Relationships among items within these bibliographic families differ, but the average family size for those works with derivative works was found to be 3.54 items. Additionally, "canonical" works that have strong cultural meaning and influence, such as Shakespeare's plays and the Bible, have very large and complex bibliographic families.[52]

5.8.2.1 Tillett's Taxonomy and FRBR

Barbara Tillett , in a study of 19th and 20th century catalog rules, found that many different catalog rules have existed over time to describe bibliographic relationships. She developed a taxonomy of bibliographic relationships that includes equivalence, derivative, descriptive, whole-part, accompanying, sequential or chronological, and shared characteristic. These relationship types span the relationship perspectives defined in this chapter; equivalence, derivative, and description are semantic types; whole-part and accompanying are part semantic and part structural types; sequential or chronological are part lexical and part structural types; and shared characteristics are part semantic and part lexical types. [53]

Smiraglia expanded on Tillett's derivative relationship to create seven subtypes simultaneous derivations, successive derivations, translations, amplifications, extractions, adaptations, and performances.[54]

In Section 3.3.2, "Identity and Bibliographic Resources," we briefly mentioned the four-level abstraction hierarchy for resources introduced in the Functional Requirements for Bibliographic Records report. FRBR was highly influenced by Tillett's studies of bibliographic relationships, and prescribes how the relationships among resources at

different levels are to be expressed (work-work, expression-expression, work-expression, expression-manifestation, and so on).

5.8.2.2 Resource Description and Access (RDA)

Many cataloging researchers have recognized that online catalogs do not do a very good job of encoding bibliographic relationships among items, both due to catalog display design and to the limitations of how information is organized within catalog records.[55] Author name authority databases, for example, provide information for variant author names, which can be very important in finding all of the works by a single author, but this information is not held within a catalog record. Similarly, MARC records can be formatted and displayed in web library catalogs, but the data within the records are not available for re-use, re-purposing, or re-arranging by researchers, patrons, or librarians.

The Resource Description and Access (RDA) next-generation cataloging rules are attempting to bring together disconnected resource descriptions to provide more complete and interconnected data about works, authors, publications, publishers, and subjects. RDA utilizes RDF to declare and store relationships among bibliographic materials.[56]

5.8.2.3 RDA and the Semantic Web

The move in RDA to encode bibliographic data in RDF stems from the desire to make library catalog data more web accessible. As web-based data mash-ups, application programming interfaces (APIs), and web searching are becoming ubiquitous and expected, library data are becoming increasingly isolated. The developers of RDA see RDF as the means for making library data more widely available online. [57]

In addition to simply making library data more web accessible, RDA seeks to leverage the distributed nature of the Semantic Web. Once rules for describing resources, and the relationships between them, are declared in RDF syntax and made publicly available, the rules themselves can be mixed and mashed up. Creators of information systems that use RDF can choose elements from any RDF schema. For example, we can use the Dublin Core metadata schema (which has been aligned with the RDF model) and the Friend of a Friend schema (a schema to describe people and the relationships between them) to create a set of metadata elements about a journal article that goes beyond the standard bibliographic information.

RDA's process of moving to RDF is well underway.[58]

5.8.3 Integration and Interoperability

Comparing and combining information between different systems involves identifying corresponding components and relationships in each system and possibly transforming them to some degree of equivalent meaning. For example, an Internet shopping site might present customers with a product catalog whose items come from a variety of manufactures who describe the same products in different ways. Likewise, the end-to-end process from customer ordering to delivery requires that customer, product and payment information pass through the information systems of different firms. Creating the

necessary information mappings and transformations is tedious or even impossible if the components and relationships among them aren't formally specified for each system

In contrast, when these models exist as data or document schemas or as classes in programming languages, identifying and exploiting the relationships between the information in different systems to achieve interoperability or to merge different classification systems can often be completely automated. Because of the substantial economic benefits to governments, businesses, and their customers of more efficient information integration and exchange, efforts to standardize these information models are important in numerous industries. Chapter 9 will dive deeper into interoperability issues, especially those that arise in business contexts.

5.9 Key Points in Chapter Five

- A **relationship** is "an association among several things, with that association having a particular significance""
- Just identifying the resources involved is not enough because several different relationships can exist among the same resources
- Most relationships between resources can be expressed using a subject-predicate-object model
- For a computer to understand relational expressions, it needs a computer-processable representation of the relationships among words and meanings that makes every important semantic assumption and property precise and explicit
- Three broad categories of semantic relationships are **inclusion**, **attribution**, and **possession**
- A set of interconnected class inclusion relationships creates a hierarchy called a **taxonomy**
- **Classification** is a class inclusion relationship between an instance and a class
- Ordering and inclusion relationships are inherently transitive, enabling inferences about class membership and properties
- Class inclusion relationships form a framework to which other kinds of relationships attach, creating a network of relationships called an **ontology**
- When words encode the semantic distinctions expressed by class inclusion, the more specific class is called the **hyponym;,** the more general class is the **hypernym**
- A **thesaurus** uses lexical relationships to suggest which terms to use
- Morphological analysis of how words in a language are created from smaller units is heavily used in text processing
- Many types of resources have internal structure in additional to their structural relationships with other resources
- The XPath language defines the structures and patterns in XML documents used by XML forms, queries, and transformations
- Many hypertext links are purely structural because there is no explicit representation of the reason for the relationship.
- Using the pattern of links between documents to understand the structure of knowledge and of the intellectual community that creates it is not a new idea

- The essential technologies for making the web more semantic and relationships among web resources more explicit are XML, RDF, and OWL
- Much of our thinking about relationships in organizing systems for information comes from the domain of bibliographic cataloguing of library resources and the related areas of classification systems and descriptive thesauri
- The Resource Description and Access (RDA) next-generation cataloging rules are attempting to bring together disconnected resource descriptions

---

[1] [Citation] The Simpsons TV show began in 1989 and is now the longest running scripted TV show ever. The official web site is www.thesimpsons.com. Yes, we know that Bart actually calls his father by his first name, but that would mess up our example here.

[2] [Citation] kinship studies

[3] [Citation] Kent's "Data and Reality" was first published in 1978 with a second edition in 1998. Kent was a well-known and well-liked researcher in data modeling at IBM, and his book became a cult classic. In 2012, seven years after Kent's death, a third edition (Kent and Hoberman, 2012) came out, slightly revised and annotated but containing essentially the same content as the book from 34 years earlier because its key issues about data modeling are timeless.

[4] [CogSci] "Semantic" is usually defined as "relating to meaning or language" and that doesn't seem helpful here. Saying that Homer is married to Marge is a semantic assertion, but saying that Homer is standing next to Marge is not.

[5] [Law]This book is not the place for the debate over the definition of marriage. We aren't bigots; we just don't need this discussion here. If this upsets you here, you will feel better in Section 5.6.1.

[6] [Citation] Winston, Chaffin, and Herman (1987)

[7] [Citation] Storey (1993).

[8] [Citation] Martin in the animated gecko who is the advertising spokesman for Geico Insurance (http://www.geico.com/). Martin's wit and cockney accent make him engaging and memorable, and a few years ago he was voted the favorite advertising icon in the US.

[9] [Citation] Gentner (1983).

[10] [Citation] Miller and Johnson-Laird, 1977, p 565.

[11] [CogSci] An example of transitivity in meronymic relationships is: (1) the carburetor is part of the engine, (2) the engine is part of the car, (3) therefore, the carburetor is part of the car. Some people have argued that meronomy isn't transitive, but a closer look at their supposed counter-examples suggests that they have confused different types of meronymic relationships. See Section 5 in Winston, Chaffin, and Herman (1987).

[12] [Computing] "Ontology" is a branch of philosophy concerned with what exists in reality and the general features and relations of whatever that might be (Hofweber, 2009). Computer science has adopted "ontology" to refer to any computer-processable resource that represents the relationships among words and meanings in some knowledge domain (Guarino, 1998).

[13] [Citation] OWL W3C

[14] [Citation] http://www.cyc.com/

[15] [CogSci] Languages and cultures differ in how they distinguish and describe kinship, so Bart might find the system of family organization easier to master in some countries and cultures and more difficult in others.

[16] [CogSci] Not quite. depends on how we define "word" - polar bear and sea horse aren't lexicalized but they are a single meaning bearing unit because we don't decompose and reassemble meaning from the two separate words. These "lexical gaps" differ from language to language, whereas "conceptual gaps" -- the things we can't think of -- may be innate and universal. things we can't perceive or experience, for example, like the pull of gravity, sort of a backwards argument, but making it on the basis that once a concept is expressed in one language it can be translated into other languages. We revisit this issue as "linguistic relativity" in chapter 6.

[17] [Citation] Example comes from Fellbaum (2010), pages 236-237. German has a word *Kufenfahrzeug* for vehicle on runners

[18] [Citation] Miller, Nouns in Wordnet (1990)

[19] [Citation] Bolshakov and A. Gelbukh (2004), p, 314. the quote continues "The references to "some class" and to "insignificant change" make this definition rather vague, but we are not aware of any significantly

stricter definition. “Hence the creation of synonymy dictionaries, which are known to be quite large, is rather a matter of art and insight”

[20] [Citation] Miller

[21] [Business] This navigation is easiest to carry out using the commercial product called “The Visual Thesaurus” at http://www.visualthesaurus.com/

[22] [Computing] Many techniques have been used here. This is a simplistic summary of all them. See Budanitsky and Hirst, Semantic distance in WordNet, an experiment.

[23] [Citation] Gross and Miller., adjectives in WordNet

[24] [Cogsci] this is called markedness

[25] [Citation] http://databases.unesco.org/thesaurus/ AAT

[26] [CogSci] Languages differ a great deal in morphological complexity and in the nature of their morphological mechanisms Mandarin Chinese has relatively few morphemes and few grammatical inflections, which leads to a huge number of homophones. English is pretty average on this scale.

[27] [LIS] These so-called endocentric compounds essentially mean what the morphemes would have meant separately. But if a ‘”doghouse” is a “dog house,” what is gained by creating a new word? This question has long been debated in subject classification, where it is framed as the contrast between “pre-coordination” and “post-coordination.” For example, is it better to pre-classify some resources as about “Sports Gambling” or should such resources be found by intersecting those classified as about “Sports” and about “Gambling.” See Svenonius (2000), pages 187-192.

[28] [Citation] Aronoff. Morphology article in MIT Encyclopedia of the Cognitive Sciences.

[29] [Citation] Six degrees of separation research, Kevin bacon game

[30] [Citation] Which seems like a ripoff of the title from a 1967 Jimi Hendrix song, “Third Stone from the Sun” - http://www.youtube.com/watch?v=5_FLmGUT_5Q

[31] [Citation]XPath – W3C and Holman

[32] [Citation] These layout and typographic conventions are well known to graphic designers but are also fodder for more academic treatment in studies of visual language or semiotics (Willams, 1994; Crow, 2010).

[33] [Computing] More going on in page rank than this, but you get the idea

[34] [Citation] Bush As We May Think 1945, p. x

[35] [Citation] Trigg and others developed extensive taxonomies of link types during the pre-web hypertext era.

[36] [Citation] Mann and Thompson 1988. For example, an author might use “See” as in “See Glushko, Mayernik, & Pepe, 2012” when referring to this chapter if it is consistent with his point of view. On the other hand, that same author could use “but” as a contrasting citation signal, writing “But see Glushko, Mayernik, & Pepe, 2012” to express the relationship that the chapter disagrees with him.

[37] [Computing] Dexter, other reference models always allowed for this, and many early research hypertext systems used them. The web dumbed down the architecture and made all links untyped, Some ambiguity in use of term binary – one-to-one vs one-to-many is a cardinality distinction, some people reserve binary to discussion about degree., See 5.6

[38] [Citation] See Weinreich, Obendorf, and Lamersdorf (2001) for a historical review of link marker conventions and some suggestions for new ones….

[39] [Computing] Probets et al (1998, Wilde (2002),

[40] [Citation] Nelson Literary Machines (1981)

[41] [Citation] Kilgour. (1999). This use of hypertext links as interaction controls is the modern manifestation of cross references between textual commentary and illustrations in books, a structural mechanism that dates from the 1500s

[42] [Computing] this is REST

[43] [Computing] Classic and well written paper Agrawal et al 1989

[44] [LIS] Garfield and de Sola Price (1963) fathers of the field… Impact factors, H-index… temporal structure of citation indicates hardness of the discipline, reliance on new work vs old work. On invisible college see Lievrouw (1989)

[45] [Law] Shepard first put adhesive stickers into case books, then published lists of cases and their citations. Now a big business forlexisnezis, westlaw (who calls it key cite).

[46] [Computing] Michael Wu etc

[47] [Cite] Chomsky syntactic structures

[48] [Computing] Berners-Lee, Hendler, and Lassila (2001) is the classic paper. See Shadbolt, Hall, and Berners-Lee (2006) for a bit of revisionist history. Somewhat ironically, the reason the Web was not semantic from the beginning was because Berners-Lee made a conscious decision to implement web documents using HTML, a presentation-oriented markup language, rather than require markup to be content-oriented. Designing HTML to be conceptually simple and easy to implement rather than general and powerful led to its rapid adoption after invention of graphical browser in 1994.

[49] [Computing] For example, Protégé (http://protege.stanford.edu/) -- a free, open-source platform with a suite of tools to construct domain models and knowledge-based applications with ontologies

[50] [Citation] http://linkeddata.org/. Wilde papers

[51] [LIS} lots of history here… some good sources are… Ad hoc techniques devised for describing the items in ancient libraries progressively became more robust and systematic over time. By the late 19th century numerous sets of cataloguing rules had been proposed that standardized author and title names and their uses in bibliographic descriptions. More importantly, the scope of descriptions was expanded to include cross references to related items. Cross references dramatically increase the value of the library catalog, because it is no longer just an aid to finding a specific item like a particular edition of Shakespeare's "Macbeth" play. Instead, the network or web of relationships between items conveys the scope and depth of the library's contents by bringing together everything that relates to the same abstract work. In addition, cross references in subject classifications assigned to bibliographic entities express the conceptual structure of the domains of knowledge represented in the collection.

[52][Citation] Smiraglia and Leazer (1999)

[53] [Citation] Tillett 1991, 1992

[54] [Citation] Smiraglia 1994.

[55] [Citation] Tillett 2005

[56] [LIS] need authoritative RDA citations here… Coyle 2010 – RDA in RDF.

[57] [Citation] (Coyle papers. 2010 – changing the nature of library data, library data in a modern context)

[58] [LIS] The FRBR entities, RDA data elements, and RDA value vocabularies have been defined in alignment with RDF using the Simple Knowledge Organization System (SKOS, 2010). The SKOS is a "RDF-compliant language specifically designed for term lists and thesauri" (Coyle, 2010c). The SKOS web site provides lists of registered RDF metadata schemas and vocabularies. From these, information system designers can create application profiles for their resources, selecting elements from multiple schemas, including the FRBR and RDA vocabularies.