Concept Maps for Data Modeling

© Ron McFadyen
Preface

Our ultimate goal is to learn to develop and document database designs, understand how to transform a design into a relational database, and know we have a highly normalized database design. When we design a database we begin by developing a model. We incorporate two modeling techniques: concept maps and entity-relationship diagrams.

Concept maps were developed as a way of capturing knowledge. Concept maps are a general-purpose tool that can be used by anyone to express concepts and their relationships using only two constructs (concepts and linking phrases). It is not difficult at all to master the technique, and you will find that software tools for creating concept maps are quite easy to use. However, it is not necessarily easy, and it may take several iterations, to create a map that you are satisfied with. Typically, concept maps are expressed as a diagram, but they are equivalent to a set of propositional statements.

To the student, concept maps are useful as an introduction to database design. To the practicing data analyst, concept maps are useful:

- When the database design phase (of a system development project) is just beginning. A design can begin as a concept map and evolve to use a more formal technique such as entity-relationship diagramming.
- When it is necessary to present models at a very high level, as in a corporate data model, where detail is not shown.

Entity-relationship diagramming is a more formal modeling technique used by data analysts to capture knowledge about data we intend to store in a database. Concept maps and entity-relationship diagrams can both be used to design databases, but entity-relationship diagrams express the same information more succinctly. Entity-relationship diagramming involves a number of specific symbols, each of which carries specific meaning.

Most computing applications will use a database for the storage and retrieval of information and relational databases are the most common form. A computer or information science student needs to understand the relational paradigm. In practice, the traditional approach to systems analysis involves the construction of an entity-relationship diagram which is subsequently mapped to a relational database. Although many database design tools can do that translation for you, the database developer must be familiar with the mapping process as sometimes an alternate mapping must be chosen.

A relational database comprises relations (in practice relations are called tables). For standard applications where a database supports the running of some organization, we expect the database to comprise normalized relations. Normalized relations will help ensure efficient applications and use of the database. The database developer needs to understand various normal forms – we will cover 1st, 2nd, 3rd, and Boyce-Codd normal forms. If it is determined that some relation is not properly normalized we can follow a
normalization process to improve the database structure. Normalization is based on the idea of functional dependency.

1. Concept Maps

1.1. Introduction

Concept maps were introduced by Joseph Novak in 1972 as a technique for representing areas of knowledge. Concept maps are diagrams that can be created easily with paper and pen, but software tools exist, such as IHMC’s CmapTools, that run on Macs and PCs. For example, see http://cmap.ihmc.us/download/. Some tools that are similar to concept maps are mind maps and idea maps; the interested student can consult the references for more information.

Some areas where concept maps are used:

- A brainstorming session may involve concept mapping to stimulate the generation of ideas.
- A student can use a concept map as a study aid for a course.
- When writing an essay, students can use a concept map to help organize the ideas they want to express.
- Concept maps can be used to visualize the changes in knowledge structure as a student progresses through a course of study.
- A systems analyst can use a concept map to capture things of interest for a computerized system (the subject of this chapter).
- A concept map can serve as an index to a collection of web pages (see http://cmap.ihmc.us/). Clicking on a node in the concept map gives access to a collection of web pages.

A concept map is a diagram constructed with concepts and links. Concepts are things or events that occur with some regularity in the area of interest. The notion of classifying and categorizing things dates back to the times of Aristotle and Plato. It’s a way for us to organize our thinking and make sense of the world around us.

Links are phrases or verbs that express a relationship between two or more concepts. For example, consider the concept table. Thinking of a table may bring to mind pieces of furniture with four legs and smooth flat surfaces suitable for eating meals. The concept chair may bring to mind furniture having a seat and back suitable for one person. The concept of human being is something we all comprehend and we are all capable of listing examples of human being: Aristotle, Plato, yourself.

Consider the concept map in Figure 1 that could be used to help explain the nature of water. This map contains a number of relevant concepts such as water, living things, plants, molecules, etc. and shows the relationships amongst concepts. Generally the diagram is read from top to bottom. For readability and understanding, a link can be
shown as a directed link. For example, the link from motion to state is shown as a directed line so that one knows to read it as motion determines state. Note how a linking phrases link can be used to link than two concepts.

![Figure 1. A concept map for the nature of water.](image)

For another example we considered a chapter from a computing text on programming with the Alice programming language - a chapter concerned with the repeated execution of statements in programs. From the introduction to the chapter it was apparent that two control structures used to control the repeated execution of instructions and methods are the loop statement and the while statement. In this chapter of that Alice text some of the obviously important concepts are: program, control structure, loop statement, while statement. Consider the following map:

![Figure 2. A concept map for Alice programming with repetition.](image)
Exercises.
1. Consider a chapter of a textbook. Create a list of the important concepts the chapter covers and then create a concept map where you use linking phrases to join concepts.
2. Consider a course that you are taking. Create a list of the most important concepts in that course. Now, what linking phrases describe the relationships amongst the concepts?

We are all probably familiar with the concepts of student and course and we might consider linking these two concepts by stating that a student registers for a course. The phrase registers for is an example of a linking phrase used to link concepts in a meaningful way. In a diagram we typically show the concepts in bubbles with a labeled line connecting concepts (see below).

If we feel it may help the readability of the map, we can make the line a directed line.

The statement a student registers for a course is called a proposition. Propositions contain two or more concepts connected using linking words or phrases to form a meaningful statement. Thus, we can see that a concept map is a diagram representing a collection of propositions. It may be easier for you to view and comprehend a large concept map with its concepts and links than it is to read the equivalent set of propositional statements.

Exercises:
3. Write out the propositional statements corresponding to each of the concept maps you created in the previous exercises.

Concept maps are developed iteratively: you may need to include more concepts, delete some concepts, and modify the way you link them several times before you are satisfied with the propositions you have formulated. As you work with concept maps, you will find that you need to be careful about choosing linking phrases in order to reduce or eliminate ambiguity. If you are sharing your concept maps with others, or using them for specific purposes such as database design, your choice of linking phrase influences your readers’ interpretations. Later on we discuss and recommend specific phrases that are useful when concept maps are intended for information modeling.

Consider the student who is studying the Alice programming language and has constructed the map we see in Figure 2. The student needs to assimilate the information from other chapters into the map. Alice has other control structures besides loop and while, and so after some thought our student creates the map in Figure 3 to explain more about Alice programming.
This diagram, created by our hypothetical Alice student, is a representation of knowledge the student is acquiring about Alice. As the Alice course proceeds the student would revisit the diagram modifying it, correcting it, or extending it in other directions. Figure 3 has captured a number of points about Alice programming:

- Important **concepts** related to Alice programs are: control structure, do together, for all in order, if, while, for all together, loop, and do in order.
- The student has expressed propositions that relate the various concepts. These **propositions** are statements linking pertinent concepts:
  - An Alice program *may use* control structures
  - Control structures *include* do together, for all in order, if, while, for all together, loop, and do in order.

Other students taking the same course may have different, but yet probably similar maps. Each student develops a map to represent their understanding, and so one could compare different maps and judge them for completeness and correctness. A student’s map is likely different from that of the instructor, but it is expected that over time the student’s concept map becomes more and more like the instructor’s map. An exercise an instructor may give to a student could be one where the student is given a collection of relevant concepts and is then asked to create the appropriate links.
A concept map contains concepts the mapmaker considers significant for the area of focus. Each proposition contained in a map expresses a way that concepts are seen to be related; each proposition can be expressed as a natural language statement involving concepts and linking phrases.

1.2. How to Construct a Concept Map

The following is adapted from the article *The Theory Underlying Concept Maps and How to Construct and Use Them* by Joseph D. Novak and Alberto J. Cañas.

- Since concept map structures are dependent on the context in which they will be used, it is best to identify a segment of a text, a laboratory or field activity, or a particular problem or question that one is trying to understand. This creates a context that will help to determine the hierarchical structure of the concept map. A good way to define the context for a concept map is to construct a Focus Question, that is, a question that clearly specifies the problem or issue the concept map should help to resolve. Every concept map responds to a focus question, and a good focus question can lead to a much richer concept map.

- Given a selected domain and a defined question or problem in this domain, the next step is to identify the key concepts that apply to this domain. These concepts could be listed, and then from this list a rank ordered list should be established from the most general, most inclusive concept, for this particular problem or situation at the top of the list, to the most specific, least general concept at the bottom of the list. Although this rank order may be only approximate, it helps to begin the process of map construction. We refer to the list of concepts as a parking lot, since we will move these concepts into the concept map as we determine where they fit in. Some concepts may remain in the parking lot as the map is completed if the mapmaker sees no good connection for these with other concepts in the map.

- The next step is to construct a preliminary concept map. This can be done by writing all of the concepts on Post-its™, or by using a computer software program. Post-its allow a group to work on a whiteboard or butcher paper and to move concepts around easily. This is necessary as one begins to struggle with the process of building a good hierarchical organization.

- It is important to recognize that a concept map is never finished. After a preliminary map is constructed, it is always necessary to revise this map. Other concepts can be added. Good maps usually result from three to many revisions.

- Once the preliminary map is built, cross-links should be sought. These are links between concepts in different segments or domains of knowledge on the map that help to illustrate how these domains are related to one another. Cross-links are important in order to show that the learner understands the relationships between the sub-domains in the map.

- It is important to recognize that all concepts are in some way related to one another. Therefore, it is necessary to be selective in identifying cross-links, and to be as precise as possible in identifying linking words that connect concepts.
addition, one should avoid “sentences in the boxes”, that is, full sentences used as concepts, since this usually indicates that a whole subsection of the map could be constructed from the statement in the box.

- If you find it is hard to add linking words onto the “lines” of a concept map, it may be because the relationships between the concepts, or the meanings of the concepts, are poorly understood. Once students begin to focus-in on good linking words, and on the identification of good cross-links, they can see that every concept could be related to every other concept. This also produces some frustration, and they must choose to identify the most prominent and most useful cross-links.

- Finally, the map should be revised, concepts re-positioned in ways that lend to clarity and better over-all structure, and a “final” map prepared.

How would you present the above procedure as a concept map? Figure 4 is a concept map taken from the publicly available concept maps at www.ihmc.org. Note how in this case the author of the concept map used links to suggest the procedural ordering.

![Figure 4. A concept map for constructing concept maps.](image-url)
Exercises.

4. If you have a procedure that you follow to write computer programs, how would you present it as a concept map?

5. If you have studied, or are studying now, a programming language, consider constructing a concept map containing about 15 or so concepts you think are most important.

6. If you have studied a computer programming language and control structures before, what control structures does that language have? Does it include the same control structures as Alice?

1.3. Hierarchies

In the section, How To Construct A Concept Map, there is mention of hierarchical structures. Hierarchies are simple structures that happen to be pervasive in computing science and its important for us to have a clear understanding of what hierarchies are.

Consider Figure 1 again that presents a concept map for the nature of water. Figure 5 reproduces this but shows directed links and with the proposition “motion determines state” removed. What is left, then, in this figure is a hierarchical structure of concepts that is said to be rooted at the concept “water”. Note that towards the bottom of the figure there are concepts that do not lead to other concepts. This is the case for oak, dog, motion, steam, lake, snow and ice. We say these concepts are leaves of the tree.

We say a concept hierarchy is a hierarchical structure of concepts arranged at several levels. The properties of hierarchies are:

- Each concept in a hierarchy has at most one linking phrase directed towards it.
- There must be one special concept (called the root concept) in the hierarchy that has no linking phrase directed toward it.
We think of hierarchies as being organized into levels. Level 0 is the level containing only the root concept. For concepts at the same level there are linking phrases that are directed to concepts at the next level. In general, we can define level \( i \) as follows: the concepts in level \( i \) are concepts such that there is a linking phrase directed to them from concepts at level \( i-1 \), where \( i>0 \). In Figure 5, level 0 comprises water; level 1 comprises living things, molecules, state; level 2 comprises plants, animals, motion, solid, liquid, gas. Level 3 comprises all the examples: oak, dog, steam, lake, snow, ice.

If you follow from one concept to another using links we say you are following a path from one concept to another. In this way you can traverse the concepts and levels of a hierarchy going from concept to concept. To traverse from the root to a specific concept there is exactly one path.

Later on it will be useful to distinguish between different types of hierarchies; for instance, a class or type hierarchy has its own properties of interest. Another term often used as a synonym for hierarchy is tree – many computing texts will include material on tree data structures. Notice how the leaves of the tree are not all at the same level. In this case we also say this is a ragged tree (hierarchy) as opposed to a balanced tree (hierarchy).

Exercises:

7. Consider Figure 5. What is the path from the root to animals? What is the path from the root to ice?
8. Consider one of your textbooks. Does it have a hierarchical structure to it?
9. Figures 2 and 3 are also examples of hierarchies. What levels are present and what concepts are in each level?

1.4. Cross-links

The section How To Construct A Concept Map mentions cross-links as something to place in a concept map once your hierarchy is established. Figure 1 contains one example of a cross-link represented by the proposition motion determines state. Figure 1 illustrates a concept map that is not a hierarchy. Taken as a whole, Figure 1 is not a hierarchical structure because there are two links leading to the concept state and this violates the rule for hierarchies that there is at most one link leading to each and every concept. Figure 1 is not an example of a hierarchy, rather it is an example of a more general structure that is referred to as a graph or network.

Exercise:

10. Consider one of your textbooks again and construct a hierarchy of chapters and topics. Can you relate a topic from one chapter to a topic in another chapter. Given that you can, construct the cross-link.
1.5. Concept maps for information modeling

Let us now consider the use of concept maps as an aid for creating databases. A database system is a computing system that stores and manages data organized in one or more databases. To use a database effectively we need to understand the nature of the data we will be managing. In this section we will work through some examples first and then go on to discuss linking phrases that should appear fairly frequently in such concept maps.

Consider a systems analyst who has the responsibility to analyze and design a computerized registration system for a university. The analyst would have a number of sources of information such as interviews, use cases, etc. The analyst knows that information about courses and students is required and may initially develop the following shown in Figure 6.

This map shows two hierarchies, one for course and one for student. Educational institutions must know what courses each student enrolls in. Now, suppose our analyst wants to add the idea that a student registers for a course, the analyst needs to add a cross-link to the diagram - see Figure 7. The cross-link in this case links the student hierarchy to the course hierarchy.
This concept map represents knowledge about the university. We say that each concept is given more meaning through the propositional statements that include the concept. As time passes the analyst will learn more about courses, students, and the university environment.

**Exercise:**

11. Consider including a new concept, registration system. Can you incorporate this concept and convert Figure 7 so there is only one hierarchy?

Suppose our analyst considers a certain aspect of courses – that some are reading courses and some are lecture courses. The analyst might create the following map.

![Figure 8. A concept map for library holdings](image)
The concept map represents the following propositions, the last four of which use *is*.

i) A lecture course *is* a course
ii) A reading course *is* a course
iii) Introduction to Statistics *is* a lecture course
iv) Readings in Shakespeare *is* a reading course

Consider propositions i) and ii). Certainly it may be true to say that a lecture course *is a course* and that a reading course *is a course*, but the relationship between course and lecture course, and course and reading course, could be more accurately described. Lecture courses and reading courses are types of courses the university offers to students. The mapmaker should want to make explicit the idea that one type of thing is a sub-concept of another concept and use an appropriate linking phrase for that purpose. In this case, many modelers prefer to use *is-a*, contains, or *includes* as a linking phrase. Whatever is chosen, we believe a team of developers should be consistent in their terminology and use only one of such phrases. We shall use *includes* and so our propositions are now:

- Course *includes* lecture course
- Course *includes* reading course

The last two propositions are ones where we are giving examples of concepts. When we state *Introduction to Statistics is a lecture course*, we are saying that *Introduction to Statistics* is an example of, or an instance of, the concept of *lecture course*. Instead of using *is*, many modelers prefer to be very explicit and use *is an example of*, or the abbreviation *e.g.* as in:

- *Introduction to Statistics is an example of* lecture course
- *Readings in Shakespeare is an example of* a readings course

The next diagram represents our modified concept map.

![Figure 9. The library map with improved linking phrases.](image)

When constructing a concept map we should be concerned with its clarity and readability - how easily others can correctly grasp our meaning. In the example we have eliminated the link word *is* and replaced it with *e.g.* or *includes*, in an effort to be more precise.
We now make our registration example a little more realistic by including instructors and the notion that courses are often offered many times during an academic year. An analyst may have determined these requirements from such things as use cases, interviews, etc.

Courses have a course number, a title, and a description. Each time a course is offered there is one instructor assigned to teach the course that is to start and end on specific dates. Each instructor has an employee id number and a name.

Students register to take an offering of a course. Students have a student id number and each student has a name.

Essentially, each of the nouns above will be used in some way as a concept: instructor, student, course, course number, title, description, offering, start date, end date, etc. We will connect concepts according to the propositions:

1. A course has a course number
2. A course has a title
3. A course has a description
4. An instructor is assigned to teach a course offering
5. An offering starts on a start date
6. An offering ends on an end date
7. An instructor has an employee id number
8. An instructor has a name
9. A course is given as an offering
10. A student registers for a course offering
11. A student has a name
12. A student has a student id number

Our concept map is then:

![Concept Map](image)

Figure 10. A concept map for an educational environment.

Consider Figure 10 as a preliminary concept map for the information needs of a university registration system. We will review and revise this concept map later after we have considered specific linking phrases that are appropriate for information modeling.
Databases are computing systems used for the storage, retrieval and updating of data. When a database is being designed we need to understand how things are related to one another so that we can store and manipulate data efficiently. For instance, we may store data about students, about courses, and we need to store information that relates students to the courses they have taken.

Linking phrases are very important. We need to clear about the relationships between concepts, we must be consistent in our use of linking phrases. If we can appreciate this, then the transition to entity-relationship diagrams is simplified.

The phrases we cover are listed below; later on, after you have studied entity-relationship diagramming you may be able to suggest some others that could be incorporated.

<table>
<thead>
<tr>
<th>Linking phrase</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Described by</td>
<td>One concept is used to describe another concept.</td>
</tr>
<tr>
<td>Uniquely described by</td>
<td>A concept is used to uniquely describe another concept.</td>
</tr>
<tr>
<td>Comprises</td>
<td>A concept comprises other concepts.</td>
</tr>
<tr>
<td>E.g.</td>
<td>An example is given for a concept. (To be precise, the example is an instance used to illustrate some concept, but is not a concept itself.)</td>
</tr>
<tr>
<td>Based on</td>
<td>The domain of values or instances of a concept is based on another concept.</td>
</tr>
<tr>
<td>Includes</td>
<td>Sub-concepts are included in a higher-level concept</td>
</tr>
</tbody>
</table>

1.5.1. Concepts that describe other concepts

When analyzing the requirements for an information system, we note that some concepts are simply descriptive of other concepts. For example, suppose in our educational environment each course has a title, a description, and a course number associated with it. In the context of information modeling we say a title is something that describes a course; similarly, description and course number are things that also help to describe a course. Educational institutions typically publish a printed calendar that describes the courses they offer. Title, description and course number are the sorts of things you see for each course.

Consider students now. Suppose for each student there will be information collected: the student’s id number, name and birth date. These are all cases where we say one concept is used to help describe another concept. Each instance of a course has a title and description. Each instance of student has a specific student id number, name and birth date. Consider the following concept diagram where we use described by as our linking phrase.
In Figure 11 we can say we have two examples of described by hierarchies. In a described by hierarchy, the higher-level concept is described by the lower-level concepts. An example of a course is said to be fully described by the appropriate examples of title, description, and course number concepts. Later on, when we discuss entity-relationship modeling we will know concepts such as student id number, student name, birth date, title, description, and course number as attributes. Attributes are things that describe instances. Student id number, name and birth date would be considered attributes of student.

When we consider concepts that describe other concepts, we can sometimes distinguish a concept as an identifying concept. Consider a student’s id number. If there is a rule that says no two students can have the same value for student id number, then student id number is not only something that describes a student, but it is something that uniquely describes a student. An identifier is something that we say uniquely describes an instance; no other instances can have the same identifier as any other. In a university environment no two students can have the same student number. Since a university cannot control student names, then name is a concept that merely describes the student but does not uniquely describe a student. We are assuming that a particular value of course number uniquely identifies a course. These considerations lead us to the following concept map.

![Diagram of concepts that describe other concepts]

![Diagram of concepts that uniquely describe other concepts]
Later on, when we discuss entity-relationship modeling we will know concepts such as student id number and course number as *key attributes*. In a concept map we can distinguish these by using the *uniquely described by* linking phrase.

Exercise:
12. Extend figure 12 to show some examples of concepts.

### 1.5.2. Knowledge about domains

In the past you may have encountered data stored in a database that was incorrect. Perhaps your last name was misspelled, your birthdate was wrong, or your gender was incorrect. Some errors can be prevented: if gender must be *male* or *female* then a value like *ale* would be totally unacceptable; but misspelling your last name was probably an unfortunate typing mistake.

When a system is computerized we need to ensure that values stored for attributes are appropriate. For instance, if we are recording the birth date for a student, we need to ensure that it is a proper date and that its value is reasonable (for example a date in the recent past but not some date in the future). If we are capturing this knowledge in a concept map we could link two concepts using the phrase “based on”. For example, see Figure 13.

![Concept Map](image)

**Figure 13. Using the phrase *based on* to indicate domains of valid values**

A systems analyst would say the concept 7 digit integer is a specification of the domain of values for a student number.
1.5.3. Concepts that are components of another concept

Some concepts are constructed from other concepts. Consider a student’s name that we naturally think of as comprising, say, first name, last name, and middle names. We can represent all of these concepts on our map and relate them as shown.

![Figure 14. Concepts that comprise other concepts.](image)

Figure 14 illustrates a two-level *comprises* hierarchy. Such hierarchies are important because they convey the knowledge that the higher-level concept stands for, or represents, the lower-level concepts.

**Exercises.**

13. Can the concept of birth date be broken down into some other meaningful concepts that are related via *comprises*?

14. If you register for courses at an educational institution, do the course numbers or identifiers comprise other concepts?

1.5.4. Sub-concepts

Another way of relating concepts is through the notion of sub-concept where we can say that some concept is a sub-concept of some other concept. Consider instructors in a university setting. A university will assign its full-time instructors to teach courses. It seems a fact of life for universities that they do not have enough full-time instructors and so they contract out for instructional services. Often such people they arrange contracts with are either graduate students or professionals employed in a related industry. Collectively, we shall refer to these other instructors as Sessional Instructors. If we consider some instance of an instructor, then that person will be either a full time instructor or a sessional instructor. An appropriate linking phrase for this is *includes*. 
The notion of sub-concepts involves hierarchies that are important in themselves. These types of hierarchies carry additional meaning. If we were to consider examples (instances) of instructor, those examples would also be examples of either sessional or fulltime. Conversely, if you give an example of sessional then you are automatically giving an example of an instructor. The proposition instructor teach courses is said to be inherited by the sub-concepts of instructor. Since sessional and fulltime are sub-concepts of instructor, we also know that sessionals teach courses and that fulltime teach courses.

The concepts of instructor, sessional, and fulltime along with the linking phrase includes form an includes hierarchy. Includes hierarchies have special properties:

- Instances of a concept at a higher level is an example of a concept at a lower level in the hierarchy.
- An example at a lower level is automatically an example of a concept at a higher level in the hierarchy.
- A concept at a lower level is said to participate in any link that appears at a higher level in the hierarchy.

The idea that some concept contains or includes other concepts is related to the notion of subtyping in entity-relationship modeling.

### 1.5.5. Examples

Recall that all the propositions given for a particular concept help to give that concept its meaning. In particular, one of the ways an information modeler can help complete the meaning of a concept is to give examples. Consider:
In the above example we want to express that Introduction to Statistics, Advanced Database, Data Warehousing and April 4, 2010 are examples of concepts. A concept map is considered to be a special case of a more general type of map called a graph; the concepts in our map are called nodes in a graph. In Figure 16, the examples shown are terminal nodes in the map – we say there is a line leading to them, but there are no lines emanating from these nodes.

When we are designing databases, it is important to give examples of concepts you are using. There is usually many others who need to read and understand what you design and examples are one way we can help others understand our meaning.

1.6. Summary

There are other linking phrases that we could introduce here, but the ones covered should be sufficient for going on to entity-relationship diagramming (next chapter). The following table summarizes recommendations for link phrases.

<table>
<thead>
<tr>
<th>Linking phrase</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Described by</td>
<td>One concept is used to describe another concept.</td>
</tr>
<tr>
<td>Uniquely described by</td>
<td>A concept is used to uniquely describe another concept.</td>
</tr>
<tr>
<td>Comprises</td>
<td>A concept comprises other concepts.</td>
</tr>
<tr>
<td>E.g.</td>
<td>An example is given for a concept. (To be precise, the example is an instance used to illustrate some concept, but is not a concept itself.)</td>
</tr>
<tr>
<td>Based on</td>
<td>The domain of values or instances of a concept is based on another concept.</td>
</tr>
<tr>
<td>Includes</td>
<td>Sub-concepts are included in a higher-level concept</td>
</tr>
</tbody>
</table>

To conclude this section we redraw the earlier concept map for an educational environment where we have incorporated a number of ideas about link phrase standardization. As you may anticipate, a diagram such as this can get quite large. This is unavoidable with concept maps as these are constructed using two simple constructs.

In the next chapter we will introduce entity-relationship diagramming which involves several constructs where each construct carries special meaning. Perhaps anyone can pick up a concept map and think they can understand something about what is presented;
this is not the case for an entity-relationship diagram. Entity-relationship diagrams are succinct descriptions of information requirements, and as such those diagrams require specialized training to create and interpret.

![Diagram of information modeling](image)

Figure 17. Using proper link phrases for information modeling.

1.7. References


[http://cmap.ihmc.us](http://cmap.ihmc.us)