Knowledge Objects and Mental-Models

M. David Merrill
Utah State University

Chapter prepared for David Wiley (Ed.)
"The Instructional Use of Learning Objects"

Cognitive psychology suggests that a mental-model consists of two major components: knowledge structures (schema) and processes for using this knowledge (mental operations). A major concern of instructional design is the representation and organization of subject matter content to facilitate learning. The thesis of this paper is that the careful analysis of subject matter content (knowledge) can facilitate both the external representation of knowledge for purposes of instruction (knowledge objects) and the internal representation and use of knowledge by learners (mental-models). If a student is taught a concise knowledge representation for different kinds of instructional outcomes, can the student use this representation as a meta-mental-model to facilitate their acquisition of specific mental-models.

Merrill (1987) elaborated the Gagné (1965, 1985) categories of learning assumptions as follows:

"There are different kinds of learned performance (instructional outcomes). Different instructional conditions are necessary to adequately promote a given type of learned performance. There are different types of cognitive structure associated with different types of learned performance. There are different types of cognitive processes necessary to use each type of cognitive structure to achieve a given type of learned performance."

Merrill (1987) suggested the following cardinal principles of instruction:

"The Cognitive Structure Principle. The purpose of instruction is to promote the development of that cognitive structure that is most consistent with the desired learned performance."

"The Elaboration Principle. The purpose of instruction is to promote incremental elaboration of the most appropriate cognitive structure to enable the learner to achieve increased generality and complexity in the desired learned performance."

"The Learner Guidance Principle. The purpose of instruction is to promote that active cognitive processing that best enables the learner to use the most appropriate cognitive structure in a way consistent with the desired learned performance."

"The Practice Principle. The purpose of instruction is to provide the dynamic, ongoing opportunity for monitored practice that requires the learner to demonstrate the desired learned performance, or a close approximation of it, while the instructor monitors the activity and intervenes with feedback both as to result and process."

This paper will elaborate the Cognitive Structure and Elaboration Principles.

Knowledge Structure

Instructional designers have long recognized the importance of analyzing subject matter for the purpose of facilitating learning via appropriate knowledge selection, organization, and sequence. An early, widely used set of categories was proposed by

Bloom and his associates (Bloom, et al. 1956, Krathwohl et al. 1964). Gagné (1965, 1985) proposed a taxonomy of learning objectives that found wide acceptance in the instructional design community. For each of his categories Gagné proposed unique conditions for learning based on information processing theory. The author elaborated and extended Gagné's categories in his work on Component Display Theory (Merrill, 1994).

While instructional designers tend to focus on delivery systems (especially technology) and to a lesser extent on instructional strategies and tactics, it is our hypothesis that the greatest impact on learning results from the representation and organization of the knowledge to be learned. Knowledge structure refers to the interrelationships among knowledge components. Gagné (1985) proposed a prerequisite relationship among knowledge components. For Gagné, the components of knowledge are facts (discriminations), concepts, rules, and higher order rules.

Reigeluth, Merrill, and Bunderson (1978) proposed that a prerequisite relationship among knowledge components represents only one type of knowledge structure. Adequate instruction would require other types of knowledge structures to be identified and made explicit to the learner. For them knowledge components are facts, concepts, steps (procedures) and principles. They proposed the following types of knowledge structures:

List. Lists often show no relationship among their components or there may be a simple ordering relationship such as size, chronology, etc., based on some attribute of the components of the list. A given set of knowledge components can be listed in a number of different ways.

Learning-Prerequisite[1]. This knowledge structure arranges components in a hierarchy indicating that a component lower in the hierarchy must be known before a component higher in the hierarchy can be learned.

Parts-Taxonomy. This knowledge structure arranges components in a hierarchy so that the coordinate components represent the parts of the superordinate component.

Kinds-Taxonomy[2]**.** This knowledge structure arranges components in a hierarchy such that the coordinate components represent kinds of the superordinate component.

Procedural -Prerequisite. This knowledge structure arranges the components (steps) of some activity to be performed in the order in which they must be executed. Procedural relations are often represented via a flow chart.

Procedural-Decision. In this structure alternative procedures are identified and the learner must consider a number of factors (conditions) in order to make a decision about which alternative is appropriate in a given situation.

Causal. In this structure the cause-and-effect relations among components are indicated.

[1] Gagné (1985) referred to this structure as a learning hierarchy.

[2] This structure is often called a concept hierarchy.

These knowledge structures were further elaborated in a conversation between Gagné and Merrill (Twitchell, 1990-91). The structures were identified as lists, taxonomies (kinds, parts, properties, functions), algorithms (path, decision), and causal nets (event chains, causal chains).

Knowledge Objects

Merrill and his colleagues in the ID_2 Research Group proposed a knowledge representation scheme consisting of knowledge components arranged into knowledge objects (Jones, Li, & Merrill, 1990; Merrill & ID_2 Research Group, 1993, 1996; Merrill, 1998; Merrill, in press). In the remainder of this paper we will refer to this work as Component Design Theory (CDT_2)[3].

CDT_2 suggests that almost all cognitive subject matter content (knowledge) can be represented as four types of knowledge objects. *Entities*[4] are things (objects). *Actions* are procedures that can be performed by a learner on, to, or with entities or their parts. *Processes* are events that occur often as a result of some action. *Properties* are qualitative or quantitative descriptors for entities, actions, or processes.

CDT_2 defines knowledge via the components of a knowledge object. A knowledge object and its components are a precise way to describe the content to be taught. The components of a knowledge object are a set of defined containers for information.

- The knowledge components of an *entity* name, describe, or illustrate the entity.
- The knowledge components of a *part* name, describe, or illustrate a part of an entity,
- The knowledge components of a *property* name, describe, identify a value, and identify a portrayal corresponding to this value for the property.
- The knowledge components of an *action* name and describe the action and identify the process(es) triggered by the action.
- The knowledge components of a *process* name and describe the process and identify the conditions (values of properties) and consequences (property values changed) of the execution of the process and any other process(es) triggered by the process.
- The knowledge components of a *kind* name, describe, and define via a list of property values a class of entities, activities, or processes.

This knowledge object framework (see Table 1) is the same for a wide variety of different topics within a subject matter domain, or for knowledge in different subject matter domains.

[3] Component Display Theory (CDT) is the original work that extended Gagné's categories of outcomes (See Merrill, 1994). Component Design Theory (CDT_2) is our current extension of this work and has been called Instructional Transaction Theory (ITT) and instructional design based on knowledge objects. We apologize for the proliferation of terms for this work.

[4] We adopted the word *entity* rather than the word *object* to avoid confusion with the use of the word *object* as used in object-oriented computer programming.

Entity: **Name** **Description** **Portrayal**	**Part**: Name Description Portrayal	**Property**: Name Description Value Value portrayal
Action: Name Description Process trigger	**Process**: Name Description Condition (value of property) Consequence (property value changed) Process trigger	**Kind:** Name Description Definition (list of property values)

Table 1. Major Components of Knowledge Objects

Some name or symbol identifies every entity (thing), action, process, or property. A given knowledge component may have several different names.

The description component is a default category in which the author can put information about an entity, a part of an entity, the property of an entity, an action associated with some entity or set of entities, a process associated with some entity or set of entities, or a class (kind) of entities, actions, or processes. For a given knowledge component there may be several different classes of information available, hence the description category may be subdivided into several sub components.

A portrayal is how a_learner senses the component. A given portrayal may be symbolic, verbal, graphic, video, animation, audio, or even olfactory or kinetic.

A property has a set of legal values that it can assume. These values may be discrete or continuous. Each of these values may also change the portrayal of the entity, action, or process.

An action often serves as a trigger for a process, hence one component of an activity is a pointer to the process that it triggers.

A process has one or more conditions. If the conditions are true the process executes, if one or more of the conditions are false then the process will not execute. A condition is defined as a value on some property in the knowledge object. If the property has the specified value, then the condition is true and the process executes. If the property has some value other than the specified value, then the condition is false and the process does not execute.

A process always results in some consequence. The consequence is defined as the change in the value of one or more properties. When the property is changed then the portrayal of that property is also changed.

A process can trigger another process, thus resulting in some kind of chain reaction. Hence, one component of a process is a pointer to the next process or processes in the chain.

One of the unique capabilities of human beings is the ability to conceptualize or to place entities, actions, and processes into categories. This capability seems to be part of the neural equipment furnished to human beings. One component of a knowledge object is a list of different category names that may be used to describe the varieties of the primary entity of the knowledge object. In a knowledge object a definition is identified as the name of the super-ordinate category (often the name of the principal entity of the knowledge object), a list of discriminating properties by which an instance in one category is distinguished from another instance in a different category, and the value of each discriminating property that defines a given class.

Knowledge Structures

Dijkstra and van Merriënboer (1997) proposed an integrative framework for representing knowledge. The cornerstone of their framework is a problem to solve. The framework attempts to identify different kinds of problems and their relationship. They have identified three types of problems: categorization problems, interpretation problems, and design problems. Categorization involves assigning instances to classes. Interpretation involves predicting the consequence of a process or finding faulted conditions in a process. Design involves performing a series of steps to accomplish some purpose, often creating some artifact.

Dijkstra and van Merriënboer identify three levels of performance associated with the three types of problems. Level 1 is characterized as learning by examples. In involves remembering a definition, a statement of a principle, or the steps in a procedure. It also involves identifying instances of a concept, identifying or describing a process, or identifying the correct or incorrect execution of a procedure. For level 1, examples of the solution and the procedure for reaching the solution are available as models for the learner.

Level 2 is characterized as learning by doing. It involves inventing concepts, predicting the consequence of a process or trouble shooting a process, or using a procedure to design a new artifact. For level 2 the procedure to reach the solution is given but the learner must find new solutions using the procedures given.

Level 3 is characterized as learning by exploration and experimentation. It involves inventing descriptive theories, hypothesizing and testing explanatory theories, and developing prescriptive theories for creating artifacts. For level 3, the task is to find both the process and the solution.

Each of these categories and levels correspond to relationships among the components of knowledge objects and among knowledge objects. These relationships are described by knowledge structures. This paper describes knowledge structures for problems of categorization and problems of interpretation. Problems of design are not included.

Concept Knowledge Structure

The knowledge components for a concept (kind) are name, description, and definition (a list of property values). A knowledge structure for a concept identifies the relationships among these knowledge components. Table 2 illustrates a knowledge structure for a concept.

		Property 1	Property 2	Property 3
	Coordinate Class A	Value$_1$	Value$_1$	Value$_1$
Name of superordinate class	Coordinate Class B	Value$_2$	Value$_2$	Value$_2$

	Coordinate Class C	Value$_3$	Value$_3$	Value$_3$

Table 2 Knowledge Structure for Concept.

This concept knowledge structure attempts to show the following relationships. A concept (kind) is always some subclass of another class (the superordinate class). There must always be at least two kinds or coordinate classes. Each coordinate class shares a set of properties with the superordinate class. Properties that have different values for two more of the subordinate (coordinate) classes are called discriminating properties. Not all properties are discriminating properties, only those who have different values for different coordinate classes. Class membership in a given coordinate class is determined by the set of values that the discriminating properties assume for members of this class.

Table 3 provides an instantiation of this knowledge structure for the superordinate concept *tree* and the coordinate concepts *deciduous* and *conifer*, kinds of trees. A third kind of tree is identified, one that has broad, flat leaves, that retains the leaves in the autumn and whose leaves do not change color. The question indicates that it is possible to identify a category (kind) but not know the name for this category.

		Shape of leaves	Retains leaves in Autumn	Leaves change color in Autumn
	Deciduous	Broad, flat	No	Yes
Tree	Conifer	Needle like	Yes	No
	?	Broad, flat	Yes	No

Table 3 Instantiation of Knowledge Structure for Concept.

Conceptual Networks

Conceptual networks are more complex knowledge structures. Conceptual networks are still composed of the same basic knowledge components. Table 4 illustrates a more complex conceptual structure. Note that property 1 has the same value for each of the coordinate classes A, B, and C. This is the property that determines class membership in this set of coordinate class. Property 2 further discriminates among the subordinate classes for class A, B, and C. This property defines the coordinate classes Aa, Ab, Ac, etc.

	Coordinate concepts	Coordinate concepts	Property 1	Property 2
		Concept $_{IAa}$	V_1	V_1
	Coordinate concept $_{IA}$	Concept $_{IAb}$	V_1	V_2
		Concept $_{IAc}$	V_1	V_3
		Concept I$_{IBa}$	V_2	V_1
Superordinate concept I	Coordinate concept $_{IB}$	Concept $_{IBb}$	V_2	V_2
		Concept $_{IBc}$	V_2	V_3
		Concept $_{ICa}$	V_3	V_1
	Coordinate concept $_{IC}$	Concept $_{ICb}$	V_3	V_2
		Concept $_{ICc}$	V_3	V_3

Table 4 A Complex Conceptual Network Knowledge Structure

Table 5 is an instantiation of a more complex concept network. Note that the first property distinguishes among the first level of coordinate concepts: birds, insects, and mammals. The second property distinguishes among the second level of coordinate concepts. Please note that for purposes of illustration the properties and property values are significantly over simplified.

	Coordinate concepts	Coordinate concepts	Locomotion	Source of food
		Finch	Fly	Plants
	Bird	Hawk	Fly	Animals
		Sparrow	Fly	Both
		Ant …	Crawl	Plants
Animal	Insects	Spider …	Crawl	Animals
		Bug …	Crawl	Both
		Cow …	Walk	Plants
	Mammal	Lion …	Walk	Animals
		Dog …	Walk	Both

Table 5 Instantiation of a More Complex Concept Network.

Processes and Activities

A process is knowledge about how something works. It answers the question, "What happens?" Processes are often taught at an *information-about* level. The process is sometimes demonstrated but the learner frequently has an incomplete or inaccurate mental-model of the process.

The components of a process include its name and description, a consequence that is defined as a change in a property value with the corresponding change in the portrayal of the entity (what happens?), and a set of conditions that is defined as values on properties (when?). A knowledge structure for a processes causal network is illustrated in Figure 1. This structure is called a PEAnet for Process, Entity, Activity Network. This structure is a very generic knowledge structure that can be used to represent almost any process. Processes are defined in terms of properties. A condition for a process is some value on a property. A consequence for a process is a change in the value of a property. When the value of a property of an entity changes the portrayal, either its appearance or its behavior also changes in a corresponding way.

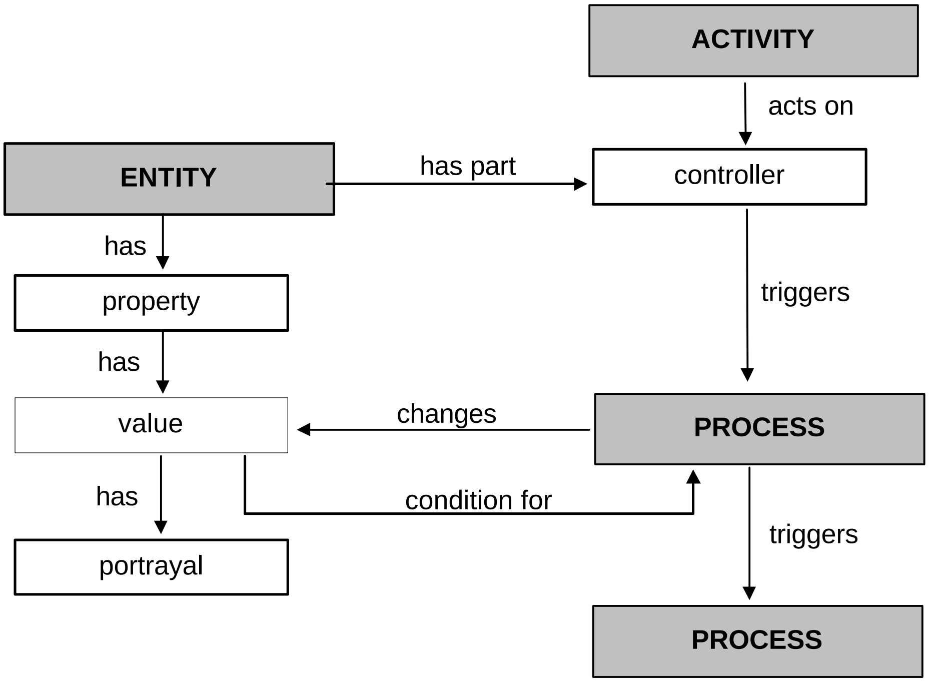


Figure 1. Knowledge Components and their Relationships in Causal Network Process Knowledge Structure.

Figure 2 is an instantiation of this PEAnet knowledge structure for the simple process of lighting a lamp when a switch is flipped. The action is for the user to *flip the switch* by moving the toggle a part of the switch. This triggers the process *change toggle position* which changes the value of the property *toggle position* from up-to-down or down-to-up, which in turn, changes the appearance of the switch as shown in the portrayals pictured. The change in toggle position also triggers another process, *light lamp*, which in turn changes the value of the *lamp lighted* property from on-to-off or from off-to-on with a corresponding change in the appearance of the lamp as depicted by the portrayals shown.

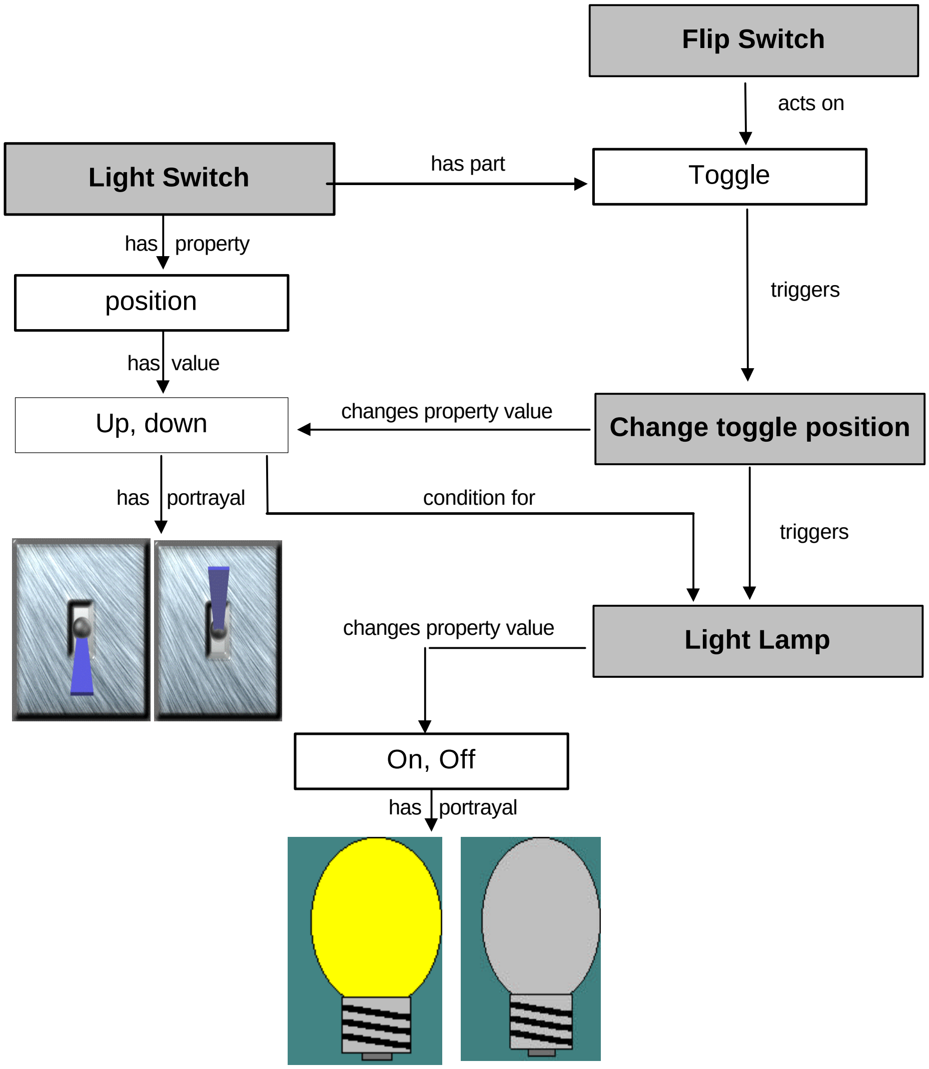


Figure 2. Instantiation of PEAnet Process Causal Network Knowledge Structure.

PEAnet knowledge structures can also be used to represent processes involving human interaction as well as devices. PEAnet knowledge structure can also be represented in table form as illustrated in Figure 3. Figure 3 and Table 6 are representations of the following situation:

A research group has acquired a new contract. Mark is being informed about the role he will play in completing this work. Three entities are involved: the manager who is performing the actions which are statements to Mark as shown in Figure 3, the boss who is merely present or absent during the conversation and Mark who is shown reacting to the news about the new contract.

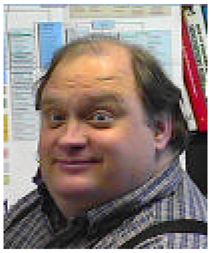
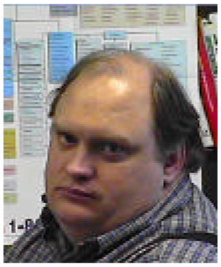
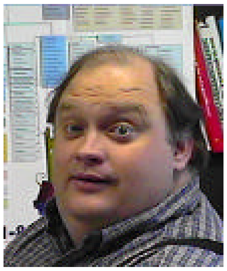
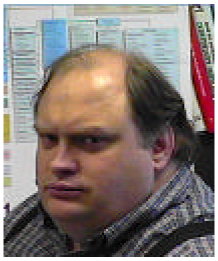
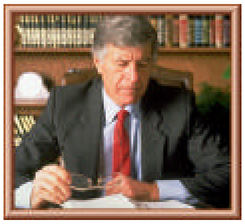
Entities	Properties	Legal Values	Portrayal
Mark	Mood	Happy	
		Sad	
		Surprised	
		Angry	
Boss	Present	Yes	
		No	

Figure 3 Entities, Associated Properties, Legal Values, and Value Portrayals for a Process Causal Network PEAnet Knowledge Structure.

Table 6 lists the actions, the process triggered by each action, the consequence of the action, and the conditions under which the consequence will occur. The actions in this PEAnet are statements of the manager to mark. The manager is not portrayed but does the actions. Note that the actions could occur in any order but in this case the usual order is as listed.

Action		**Process**		**Consequence**	**Condition**
Statement "We have a new contract."	*triggers*	Make Mark happy	*changes*	Mood = happy	
Statement "But you don't get to direct the project."	*triggers*	Make Mark sad	*changes*	Mood = sad	
Statement "Jean will direct this project"	*triggers*	Make Mark surprised	*changes*	Mood = surprised	
Statement "You get to work for Jean"	*triggers*	Make Mark angry	*changes*	Mood = surprised	Boss present = yes
				Mood = angry	Boss present = no

Table 6 PEAnet Actions, Processes, Consequences and Conditions
for a Simulation of an Office Conversation.

What happens? When the first statement (action) is executed the process *Make Mark Happy* changes the property *mood* to the value *happy*. The happy portrayal of Mark corresponding with the value happy is shown. When the next statement (action) is executed the process *Make Mark sad* changes the property *mood* to the value *sad* and the corresponding portrayal showing a sad mark is displayed. A similar chain of events occurs for the third statement. For the forth statement the process *Make Mark angry* changes the value of *mood* to the value *angry* only when the property *boss present* has a value of *no*. When the *boss present* property has a value of *yes* then the value of mood remains or is changed to *surprised*.

This is a significantly oversimplified PEAnet to illustrate the role of properties, property values, conditions, and consequences. Actions can trigger more than one process. Processes can change more than one property. There can be many or few conditions. Conditions for a process may have been set much earlier in a complex process that consists of many events.   Hopefully the reader can extrapolate to the more complex case from the information given.

Mental-models

Cognitive psychologists have proposed a variety of theories of how knowledge is represented in memory (See Mayer, 1992). Schema theory postulates that learners represent knowledge in memory as some form of cognitive structure. A knowledge structure is a form of a schema. A knowledge structure represents the information that is required if a learner is to be able to solve problems. If the required information (knowledge components) and the relationships among these knowledge components are incomplete, then the learner will not be able to efficiently and effectively solve problems requiring this knowledge.

Mental-models combine a schema or mental representation with a process for manipulating the information in the schema. Solving a problem requires the learner to not only have the appropriate knowledge representation (schema or knowledge structure) but he or she must also have algorithms or heuristics for manipulating these knowledge components in order to solve problems.

Categorization Problems

Classification

A common instructional strategy for teaching concepts (kinds-of) is to present a set of examples representing the different coordinate classes of the superordinate class. The learner is told the class membership for each example. The learner is then given a previously unencountered set of examples and asked to classify or sort each of them into the appropriate coordinate class.

Tables 2 and 3 illustrate a knowledge structure for a set of coordinate concepts. The knowledge components and relationships in this knowledge structure are necessary if the learner is to be able to correctly classify new examples. The algorithm necessary for concept classification requires the learner to do the following:

- Remember or have available the properties and values associated with each category (the definition).
- For a given example, find the portrayal of each property in the portrayal of the example. Determine its value. Repeat for each of the properties required to determine class membership.
- Compare the property values of the example with the property values associated with each concept class. When a match is found then give the name of the coordinate class associated with these property values.

Having a concept knowledge structure such as that illustrated in Table 3 also allows the learner to explore "What if?" problems about the concepts under consideration. Having a schematic representation of the properties and their values allows the learner to speculate about coordinate classes that may not have been specified by the instruction. For example, in Table 3 the third row enables the learner to ask, "Are their trees that have broad leaves that don't change color and drop in the Autumn? Is so, what are they called?" The learner can also explore other combinations of property values. The schema and this cognitive exploration process enable the learner to extend and elaborate their concept understanding.

Generalization

Table 5 and Table 7 illustrate a more complex knowledge structure that enables a learner to make generalizations. A generalization is when classes from different set of coordinate concepts are seen as coordinate concepts for a new set of coordinate concepts. In Table 5 finches, ants, and cows each appear in a different coordinate set corresponding to birds, insects, and mammals. However, each of these classes share the same value, *plants*, on property 2, *Source of Food*. By sorting on the second property we can identify a new set of coordinate concepts as in Table 7. Note in this case that the first property discriminates on the second set of coordinate concepts while the second property discriminates on the first level of coordinate concepts.

	Coordinate concepts	Coordinate concepts	Locomotion	Source of food
Animal	Herbivore	Finch	Fly	Plants
		Ant	Crawl	Plants
		Cow	Walk	Plants
	Carnivore	Hawk	Fly	Animals
		Spider	Crawl	Animals
		Lion	Walk	Animals
	Omnivore	Warbler	Fly	Both
		Bug	Crawl	Both
		Dog	Walk	Both

Table 7 Instantiation of a Complex Conceptual Network Knowledge Structure

What is the algorithm by which the learner solves problems of generalization? First, if the value on a property, or the values on two or more properties, are the same across different sets of coordinate concepts, then sorting on this property will yield a new set of coordinate classes. Having a complex knowledge structure of one set of concepts allows the learner to manipulate the knowledge components in this manner in an attempt to identify other coordinate concepts. The learner then searches to find the names associated with these new coordinate concepts (Dijkstra & van Merriënboer level 2). If this is a new categorization then the learner may invent names and thus invent new concepts (Dijkstra & van Merriënboer level 3).

Interpretation Problems

Explanation

The knowledge components of property, value, portrayal, condition, and consequence provide a vocabulary by which the learner can provide a precise explanation of a given process. Having a learner determine the knowledge structure PEAnet of a given process provides a very precise way to assess the completeness and accuracy of the learner's mental-model for a given class of problems.

Prediction

What is involved in problem solving at the application level? Merely explaining a process, even with the precision of a PEAnet, is at the first level of Dijkstra and van

Merriënboer's representation of problems. Making prediction moves the learner to the learn-by-doing level. For a given situation, making a prediction involves determining first the conditions that are relevant to the consequence. This involves finding the portrayal of the property(s) involved and determining its current value(s). The learner must know the principle involved, that is, if <conditions> then <consequence>, and determine which of several such relationships is applicable in the situation under consideration. The learner can then indicate the change in property(s) value that will occur and the corresponding change in the portrayal of the entity(s) under consideration.

Trouble Shooting

Another type of problem solving at the learn-by-doing level is troubleshooting. Troubleshooting is the inverse of prediction. In this situation the learner is shown some consequence (a change in property value and its concurrent change in portrayal). The learner must then determine what property was changed. The learner must recall the relevant principles: If <condition(s)> then <consequence(s)>. He or she must match the consequence that occurs to a statement of principle. This enables the learner to identify the conditions that may be faulted. He or she must then examine the situation to find the portrayal of the potentially faulted condition(s) and determine if the value of the associated property value matches the principle. If not, this is possibly the fault. The learner can then correct the fault, that is, change the value of the property that is the faulted condition and then test to see if the desired consequence occurs.

In single event processes this troubleshooting problem solving requires little more than memory. However, in very complex processes, involving many events, the above mental process may need to be repeated many times until the faulted conditions are identified and corrected.

PEAnets provide a very specific way to define both prediction and troubleshooting. They also provide a vocabulary for use by the learner to be more precise in their problem solving activities.

Meta-Mental-Models

A meta-model is a model for a model. The knowledge structures described in this paper for concepts and processes and their associated cognitive processes for different types of problem solving using these knowledge structures provide a potentially useful meta-mental-model for a learner. If the learner has acquired the knowledge components and knowledge structure for a conceptual network, then he or she has a meta-mental-model for acquiring a conceptual network in a specific area. This meta-mental-model allows the learner to seek information for slots in the model. It provides a way for the learner to know if they have all the necessary knowledge components to instantiate their mental-model. It enables the learner to extend their model of the concept under consideration by processing the concept schema for additional classes or by processing the schema to determine potential generalizations.

If the learner has acquired the knowledge components and knowledge structure for a PEAnet then he or she has a meta-mental-model for acquiring a process mental-model for some specific phenomena. The PEAnet structure enables the learner to determine if all the necessary knowledge components are present. By representing the phenomena in a PEAnet the learner can run mental experiments to see what consequences should occur under given sets of conditions. The learner can conduct mental "what if" experiments to predict what-happens when the conditions change. The learner can represent very complex phenomena in a very systematic way providing a much better understanding of the phenomena under consideration. Furthermore, the learner can describe devices or situations that don't work correctly using the PEAnet meta-mental-model. This allows the learner to help determine why a given process is not working by identifying the conditions that may be faulted.

Automated Instructional Design

Knowledge structures also make it possible to build smart instructional systems. A knowledge structure represents a precise way to represent a conceptual network or a causal network. The processes identified for manipulating the knowledge objects in a knowledge structure provide the basis for computer algorithms that can emulate some of the processing done by a learner.

In previous papers we have described instructional simulations based on PEAnet knowledge structures. There is not space in this paper to elaborate these ideas here. (See Merrill, 1999, Merrill, in press$_b$).

Summary

This paper describes knowledge components that are thought to be appropriate and sufficient to precisely describe certain types of cognitive subject matter content (knowledge). It also describes knowledge structures that show the relationships among these knowledge components and among other knowledge objects. It suggests that a knowledge structure is a form of schema such as those that learners use to represent knowledge in memory. A mental-model is a schema plus cognitive processes for manipulating and modifying the knowledge stored in a schema. We suggested processes that enable learners to manipulate the knowledge components of conceptual network knowledge structures for purposes of classification, generalization, and concept elaboration. We further suggested processes that enable learners to manipulate the knowledge components of process knowledge structures (PEAnets) for purposes of explanation, prediction, and troubleshooting. The hypothesis of this paper is that knowledge components and knowledge structures, such as those described in this paper, could serve as meta-mental-models that would enable learners to more easily acquire conceptual and causal networks and their associated processes. The resulting specific mental-models would facilitate their ability to solve problems of conceptualization and interpretation.

References

[1] Bloom, B. S., Englehart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (Eds.) (1956). *A Taxonomy of Educational Objectives: Handbook I. Cognitive Domain.* New York: David McKay.

Dijkstra, S. & van Merriënboer, J. J. G. (1997). Plans, Procedures,and Theories to Soplve Instructional Design Problems. In S. Dijkstra, N. M. Seel, F. Schott, & R.D. Tennyson (1997). *Instructional Design: International Perspectives. Volume 2 Solving Instructional Design Problems.* Mahwah, NJ: Lawrence Erlbaum Associates.

Gagné, R. M. (1965, 1985). *The Conditions of Learning.* New York: Holt Rinehart and Winston.

Jones, M.K., Li, Z. & Merrill, M.D. (1990). Domain knowledge representation for instructional analysis. *Educational Technology*, 30(10), 7-32.

Krathwohl, D.R., Bloom, B. S., & Masia, B. B. (Eds.) (1964). *Taxonomy of Educational Objectives: Handbook II, Affective Domain.* New York: David McKay.

Mayer, R. E. (1992). *Thinking, Problem Solving, Cognition. 2nd* Ed. NY: W.H.Freeman.

Merrill, M. D. (1987). The New Component Design Theory: Instructional design for courseware authoring. *Instructional Science*, 16, 19-34.

Merrill, M. D. & ID2 Research Team (1993). Instructional Transaction Theory: knowledge relationships among processes, entities, and activities. *Educational Technology*, 33 (4), 5-16.

Merrill, M. D. (1994). *Instructional Design Theory*. Englewood Cliffs: Educational Technology Publications.

Merrill, M. D. & ID_2 Research Team (1996). Instructional Transaction Theory: Instructional Design based on Knowledge Objects. *Educational Technology, 36 (3), 30-37.*

Merrill, M. D. (1998). Knowledge Objects. *CBT Solutions*, March/April issue, pages 1, 6-11.

Merrill, M. D. (1999). Instructional transaction theory (ITT): instructional design based on knowledge objects. In C. M. Reigeluth (Ed.). Instructional Design Theories and Models:Volume II A New Paradigm of Instructional Design. Mahwah, NJ: Lawrence Erlbaum Associates.

Merrill (In $Press_a$). Components of instruction: toward a theoretical tool for instructional design. *Instructional Science.*

Merrill, M. D. (In $Press_b$). A knowledge object and mental-model approach to a physics lesson. *Educational Technology.*

Reigeluth, C. M., Merrill, M. D. & Bunderson, C. V. (1978). The structure of subject matter content and its instructional design implications. *Instructional Science*, 7(2), 107-126.

Twitchell, D. (Ed) (1990-91). Robert M. Gagné and M. David Merrill in conversation. *Educational Technology*. A series of 7 articles No.1 30(7), 34-39; No. 2 30(8), 36-41; No. 3 30(9), 36-41; No.4 30(10), 37-45; No.5 30(11) 35-39; No. 6 30(12) 35-46; No. 7 31(1)34-40.