Instructional Transaction Shells:

Responsibilities, Methods ,and Parameters

M. David Merrill, Zhongmin Li and Mark K. Jones

Utah State University

Written July 1991

Published Educational Technology, 1992, 32(2), 5-27.

INSTRUCTIONAL TRANSACTION SHELLS:

RESPONSIBILITIES, METHODS, AND PARAMETERS

M. David Merrill, Zhongmin Li and Mark K. Jones[1]

Utah State University

July 1991

ABSTRACT

In this paper we suggest that an instructional transaction must carry out four responsibilities: knowledge selection, knowledge sequence, interaction management, and interaction enactment . We have suggested that these responsibilities are accomplished via methods, which are sets of activities that enable each responsibility to be accomplished. All transactions include knowledge selection methods for partitioning knowledge, portraying knowledge, and amplifying knowledge; knowledge sequence methods for routing the learner through the selected knowledge and guiding the learners advancement; interaction management methods for prioritizing interactions and expediting the learners acquirement of the knowledge involved in the interaction; and interaction enactment methods for overviewing , presenting , enabling practice, and assessing knowledge. Finally we have suggested that the way a given method carries out its activities is determined by a set of parameter values associated with this method. We have identified a number of parameter values associated with the methods of one type of transaction, an identify transaction.

DEFINITION OF INSTRUCTIONAL TRANSACTION SHELL

In a previous papers (Li & Merrill, 1990; Merrill, Li & Jones, in press) we have defined instructional transactions as instructional algorithms, patterns of learner interactions (usually far more complex than a single display and a single response), which have been designed to enable the learner to acquire a certain kind of knowledge or skill. Different kinds of knowledge and skill would require different kinds of transactions. The necessary set of these instructional transactions are designed and programmed once, like other applications such as spread sheets and word processors. These instructional programs are called instructional transaction shells. These transaction shells can then be used with different content topics as long as these topics are of a similar kind of knowledge or skill.

A transaction shell (see Figure 1) consists of four[2] primary components: *interactions and an interaction manager* which causes the transaction to occur; *instructional parameters* which enable the instruction to be customized for a given learner population, learning task, and environmental situation; a *knowledge base* containing a structural representation of all the

---

[1] The authors are faculty members of the Instructional Technology Department at Utah State University. The authors acknowledge the contributions of their associates Ann Marie Canfield, Scott Schwab, Carolyn Thorsen, Jennifer Chen, Jinelle Monk, and graduate students in the authors' classes.

[2] This definition of a transaction shell is an elaboration of our previous definition (See Merrill, Li, & Jones, in press).

knowledge[3] to be taught; and a *resource data base* containing mediated representations of the knowledge to be taught. A transaction shell has three authoring systems: a *transaction configuration system*, a *knowledge acquisition system*, and *resource editors*. The knowledge acquisition system enables a subject matter expert to structure the knowledge to be taught. The resource editors enable the creation of mediated representations of the knowledge. The transaction configuration system enables the designer to provide values for a wide range of instructional parameters.

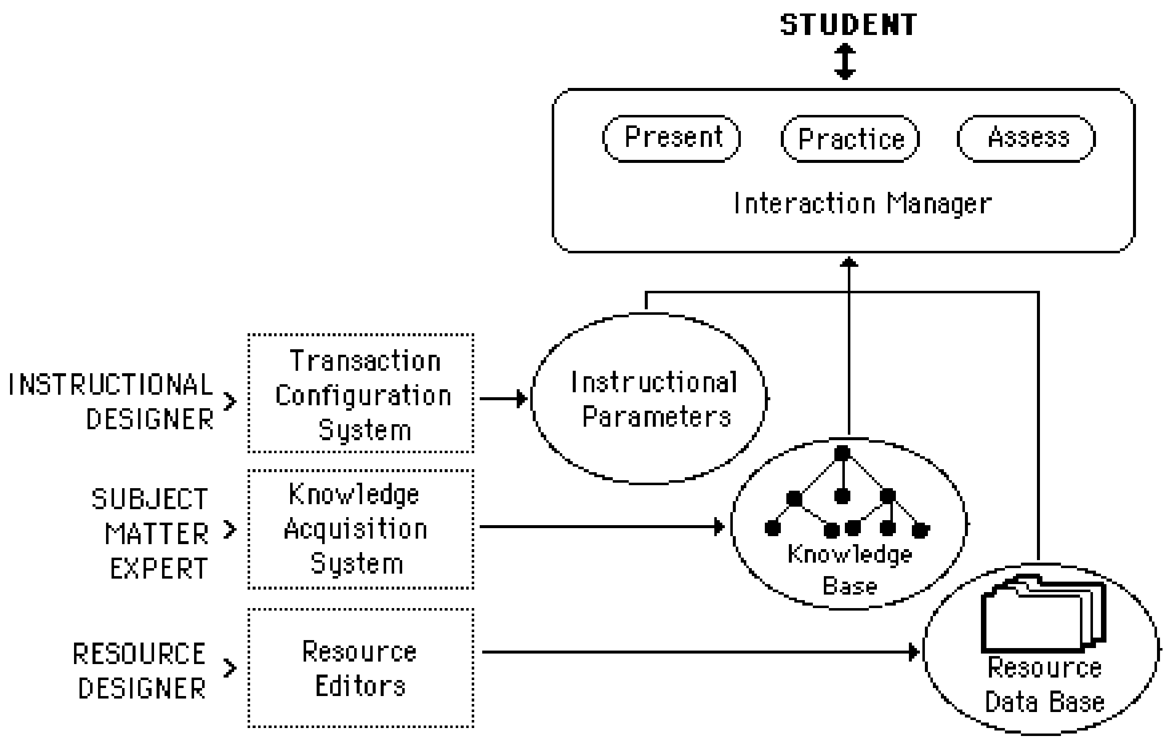


Figure 1 Components of an Instructional Transaction Shell

Transaction Shell Interactions

Adequate interactions are not passive, they require relevant mental effort on the part of the learner. An interaction can vary in terms of the level of mental effort required. Relevant mental processing causes the learner to actively construct the appropriate mental model of the target knowledge structure(s). Mere activity, or overt responding, is not necessarily relevant mental processing. Asking an appropriate rhetorical question, which requires no overt response, may require far more active mental processing than having the learner respond overtly by pushing the return key or space bar to turn the page. A transaction, in which the interactions all require the learner to engage mental processes directly related to the knowledge structure being promoted, will be more effective than one in which many of the interactions are passive or less related to the mental processes represented by the knowledge structure (See Merrill, 1988). The degree of mental effort required should not be an alternative, rather, a given transaction should be

---

[3] We will use the word *knowledge* in this context to include both declarative knowledge (information) and procedural knowledge (skills or abilities to be acquired).

defined such that its interactions require all the mental processing that is required for the learners to acquire the target knowledge structure(s).

The type of transaction and the components of its knowledge base limit the interactions that are possible within a given transaction shell. Different classes of transactions will have different types of interactions. Nevertheless, all transactions should include interactions that are characterized by certain *interaction modes*. Four interaction modes have been identified: *overview*, *presentation, practice,* and *assessment.*

Individual interactions may be primarily aimed at presenting information (expository) or they may be primarily aimed at providing an opportunity for learner practice or assessment (inquisitory). An adequate transaction should have capabilities for both expository and inquisitory interactions. That is, some interactions should focus on overview or presentation[4] while other interactions should focus on practice or assessment. Interaction alternatives are described in terms of presentation forms: expository generality (EG), expository instance (Eeg), inquisitory instance practice/assessment (Ieg), and inquisitory generality practice/assessment (IG). (see Merrill, 1983, 1987, 1988).

An adequate transaction should allow both learner and system controlled interactions. Learner control has attained wide spread popularity in recent years but indications are that not every learner can make good decisions about their own learning. A transaction needs to have the capability to direct learner activity as well as having the learner control the interaction; it needs to include both learner controlled and system controlled interactions. The degree of learner control provided in any given situation is a function of the familiarity, motivation, aptitude, and attitude of the individual learners involved.

Interaction modes are under system or learner control. Under system control (SC) the system sequences the displays and responses in the interaction mode. Under learner control (LC) the learner sequences the displays and responses. LC overview enables the learner to explore the knowledge structure(s); SC overview would systematically overview the knowledge structure(s). LC presentation would enable the learner to explore the knowledge in any order; SC presentation would systematically sequence the knowledge for the learner. LC practice would allow the learner to engage in practice opportunities in any order; SC practice would present practice opportunities one-by-one in a predetermined systematic order. Table 1 defines several types of interaction mode for system and learner control. Footnotes define the symbols and technical terms in the table.

[4] Consistent with the previous paragraph on active mental effort, Merrill (1988) pointed out that a presentation may still involve an interchange of information. Presentation does not mean mere telling. An expository presentation may still involve an active dialog with the student.

Table 1 Interaction Mode Alternatives for System and Learner Control

INTERACTION MODE[5]:	SYSTEM CONTROL	LEARNER CONTROL
OVERVIEW (KS)	System sequenced presentation of the Knowledge Structure	Learner exploration of the Knowledge Structure
PRESENTATION (E)	System sequenced presentation of EG[6] + Eeg[7]	Learner sequenced presentation of EG + Eeg
PRACTICE (I)	System sequenced presentation of Ieg[8]	Learner sequenced presentation of Ieg
G[9] ASSESSMENT (TG)	System sequenced presentation of IG[10]	Learner sequenced presentation of IG
eg[11] ASSESSMENT (Teg)	System sequenced presentation of Ieg	Learner sequenced presentation of Ieg

A transaction shell is seldom composed of only a single type of interaction. It includes the complete sequence of presentations and reactions necessary for the learner to acquire a specific type of knowledge. This sequence of interactions we will call an *interaction strategy*.

Interaction strategy is the combination and sequence of interaction modes available to the learner. We have identified at least seven types of interaction strategy: *overview, familiarity, basic, mastery, basic remediation, mastery remediation*, and *assessment.* Overview consists of an overview interaction mode. Familiarity consists of an overview interaction plus a presentation. Basic instruction consists of an overview plus presentation plus practice. Mastery instruction consists of overview plus presentation plus practice plus generality and/or instance assessment; if the criterion is not met a new presentation, practice, and assessment for missed items is engaged until the criterion is met. Basic remediation consists of generality or instance assessment; if the criterion is not met then basic instruction (see previous item) is provided for the missed items. Mastery remediation consists of generality or instance assessment; if the criterion is not met then mastery instruction (see previous item) is provided for the missed items until the criterion is met.

---

5 EG, Eeg, IG and Eeg are the Primary Presentation Forms (PPF) of Component Display Theory (CDT). Interaction modes are defined in the technical terms of CDT (See Merrill, 1983, 1987, 1988).

6 EG = Expository Generality -- presenting the general rule to the student.

7 Eeg = Expository Instance -- presenting the specific case to the student.

8 Ieg = Inquisitory Instance -- requiring the student to respond to the specific case.

9 G = Generality, the general case or rule.

10 IG = Inquisitory Generality -- requiring the student to respond to the general case.

11 eg = example or instance, the specific case.

Assessment consists of generality or instance assessment. Some transaction shells will include other strategies.

Interaction strategy is under system or learner control. Under learner control the learner is given a menu and can select any of the available interaction modes in any order. Learner control may be accompanied by advice indicating what the learner has accomplished to date and advising the learner what they should do next. Under system control the system determines which mode is next and when the transaction shifts to the next mode. Table 2 outlines the interaction strategy alternatives for system and learner control.

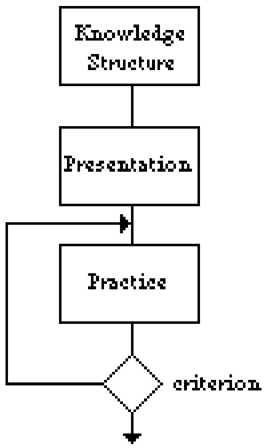
Table 2 Interaction Strategy Alternatives for System and Learner Control

INTERACTION STRATEGY[12]	SYSTEM CONTROL	LEARNER CONTROL
OVERVIEW	SCKS -- system sequenced presentation of knowledge structure.	LCKS -- learner exploration of knowledge structure.
FAMILIARITY	SCKS + SCE -- system sequenced presentation of knowledge structure plus presentation.	LCKS + LCE -- learner exploration of knowledge structure plus learner sequenced presentation.
BASIC	SCKS + SCE + SCI -- system sequenced presentation of knowledge structure plus presentation plus practice.	LCKS + LCE + SCI -- learner sequenced exploration of knowledge structure plus presentation plus practice.
MASTERY	SCKS + SCE +SCI + TG or Teg -- Basic instruction + testing.	LCKS + LCE +LCI + TG or Teg -- Basic instruction + testing.
BASIC REMEDIATION	TG or Teg + BASIC @ criterion -- testing with basic instruction used to remediate errors.	TG or Teg + BASIC @ criterion -- testing with basic instruction used to remediate errors.
MASTERY REMEDIATION	TG or Teg + MASTERY @ crit. -- testing with mastery instruction used to remediate errors	TG or Teg + MASTERY @ crit. -- testing with mastery instruction used to remediate errors
ASSESSMENT	System sequenced TG or Teg	Learner sequenced TG or Teg

[12] Interaction strategy is defined in terms of sequences of presentation modes.

The following figures indicate, via flow charts, the sequence of interaction modes for the more complex instructional strategies identified in Table 2. It should be noted that these flow charts indicate basic sequences rather than rigid strategies. By means of on-line monitoring of a given learner's performance, via an advisor function, a given strategy may be changed during the instruction enabling a wide variety of different configurations for instructional strategy. The following flow charts are merely default paths from which a given learner may vary.

Basic Instruction



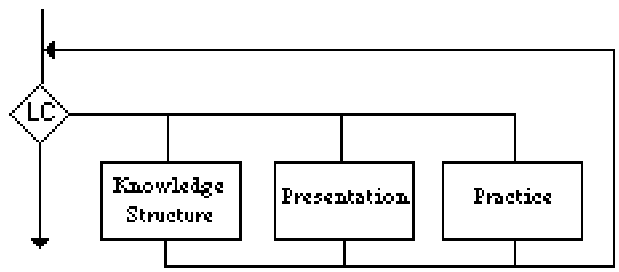
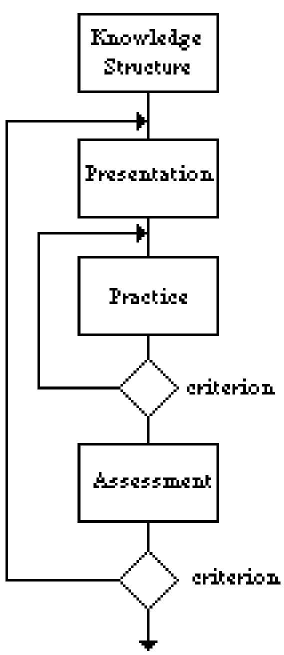


Figure 2a SC Basic Instruction Figure 2b LC Basic Instruction

Mastery Instruction



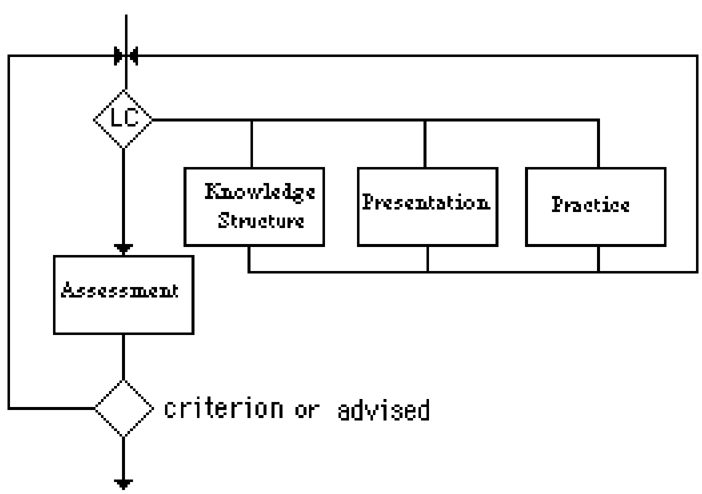
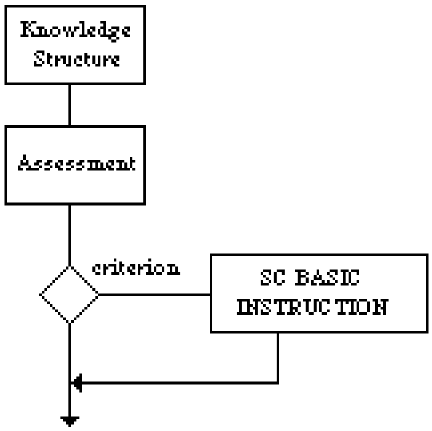


Figure 3a SC Mastery Instruction Figure 3b LC Mastery Instruction

Basic Remediation



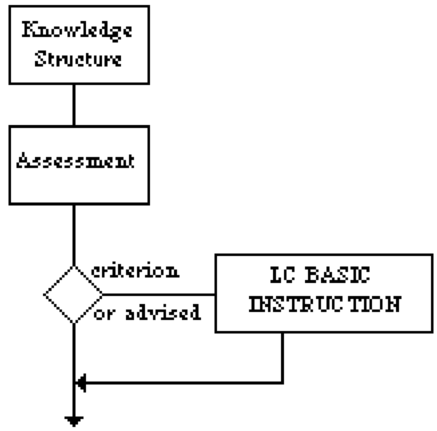
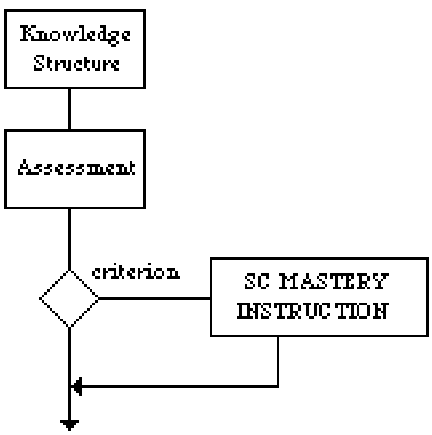


Figure 4a SC Basic Remediation Figure 4b LC Basic Remediation

Mastery Remediation



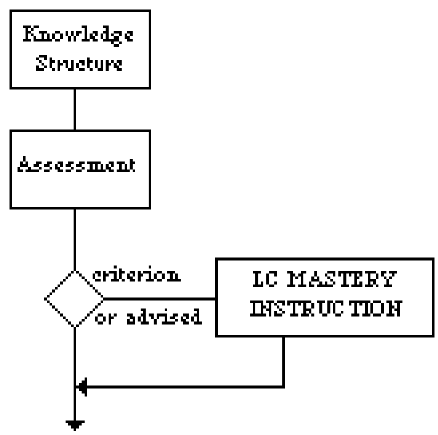


Figure 5a SC Mastery Remediation Figure 5b LC Mastery Remediation

Knowledge base and resources

The knowledge base of a transaction shell is the formal structure of the knowledge. This structure is known by a given transaction shell which is able to take advantage of this formal structure to conduct its enactment of a given transaction. This formal knowledge structure is delivery system independent. Associated with this knowledge base is a set of resources. Resources are usually represented in terms of a particular representational device and are most often delivery system dependent. A given piece of knowledge in the knowledge base may be represented by more than one resource. The transaction is able to use alternative resources since how to present this information is determined by the structure of the knowledge base not the content of the specific resources. This also enables a resource to be replaced by a more current resource as long as the links to the knowledge base are reestablished.

Figure 6 illustrates a parts cluster diagram, the formal structure of an entity frame in the knowledge base. Each box in the tree diagram indicates a part of the entity or part represented by its superordinate box. The diagram indicates how the parts are clustered in the entity being taught, i.e., what parts are parts of what other parts. The entity or part frames (represented by the boxes in the tree structure) contain slots for the name of the part or entity, the properties associated with this part or entity, the legal values that these properties may assume, and pointers to the part location on each of the primary resources associated with the frame. Slots also exist which contain pointers to various types of ancillary information -- such as its pronunciation, function, a description, or an aside note -- represented by secondary resources associated with any part or entity frame.

<u>chunk size ["n"]</u>

A cluster diagram has associated with it one knowledge base parameter, *chunk size*. Chunk size is the maximum number of entities or parts allowed in a coordinate set. The transaction shell will take advantage of chunk size in partitioning the instruction into mind sized pieces. Chunk size is usually limited to 7± 2 parts but may be varied for particular subject matters.

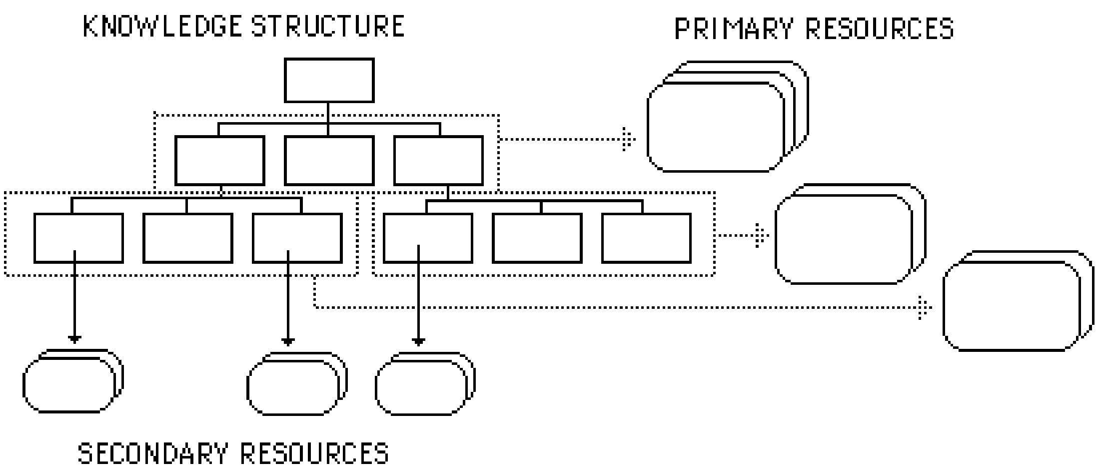


Figure 6 Relationship of knowledge base to primary and secondary resources.

Figure 6 also illustrates that a set of primary resources are associated with each *coordinate set* of components represented by the cluster diagram. These resources may be any media allowed by a particular delivery system, such as, text, a graphic, an animation, a video segment, or an audio segment. A given coordinate set may have more than one resource, but each resource must contain representations of the components identified by the coordinate set and its superordinate entity or part.

Figure 6 also indicates that secondary resources can be associated with each entity or part of the structure diagram. These secondary resources are usually audio and/or visual and for entity structures contain ancillary information such as pronunciation, function, a description, or an aside note.

The nature of the primary resource can have a significant effect on the learning that occurs with the representation of the subject matter. An adequate delivery system should be able to accommodate a wide variety of resource types -- text, graphics, animation, video, audio. Each of these resource types is characterized by a number of representation parameters: view, mode, and fidelity. A given instantiation of a transaction shell will usually include only two or three alternative resources for a given coordinate set. A very rich instructional environment may contain alternative resources representing a wide sampling of the representation parameters.

View [structural, physical, functional]

The same information is represented in a number of different ways. View indicates the principal ways that knowledge is represented in a transaction shell. A *structural* view represents the knowledge components in terms of some formal structure diagram, such as a tree diagram or a flow chart. It is usually unnecessary to include a resource showing the structural view since a transaction shell can construct this view directly from the knowledge base. A *physical* view represents the components of the entity, activity, or process in a way that shows how it looks as well as how it works. A *functional* view represents the components of the entity, activity or process in a way that shows how it works rather than how it looks. Each of these types of representation view is modified by two additional parameters: mode and fidelity .

Mode [literal, symbolic, language]

Mode refers to the nature of the physical or functional view. There are at least three primary modes: literal, symbolic, and language. *Literal* representation provides information that is perceived by one or more of the senses: sight, hearing, touch, smell and taste. The corresponding representations are visual (still and motion), auditory, tactile, olfactory and taste. Instructional delivery systems differ to the extent that they can implement literal representations. An adequate instructional delivery system should be able to implement at least visual and auditory representation. Many activities require some tactile representation. Adequate transactions for many subject areas may be seriously limited by delivery systems with limited representational capabilities. However, the capabilities of instructional delivery systems continues to expand with vendors announcing increased capabilities with increasing frequency. True multi-media systems, integrating digital representation of motion visual, graphic, text and audio have been or are about to be announced. Futuristic systems implementing virtual reality involving helmets and gloves which allow remarkable tactile involvement have been demonstrated. Literal representations can vary from these high-technology, futuristic, virtual-reality simulations to simple line drawings. The challenge for an instructional theory is to be able to prescribe those necessary and sufficient representations for particular instructional tasks. Physical representations typically involve some form of literal representation but may also be accompanied by symbolic or language representations as well.

Symbolic representation includes notation systems, diagrams, and graphs of all sorts. We have distinguished symbolic representation from language even though this is an arbitrary distinction since language is certainly symbolic. Symbolic in this context means those representations that go beyond ordinary language to provide increased precision for a particular subject matter area. Symbolic representations include music notation, map symbols, circuit symbols, architectural symbols, math symbols, etc. Symbols may have arbitrary association with their corresponding reality; they be analogous to their corresponding reality; or they may be iconic to their corresponding reality (indicated by the fidelity parameter). Functional representations typically involve some form of symbolic representation but may also be accompanied by language representations as well.

Language involves description via words and grammatical structures. Except where language or written expression are the entities involved in the instruction, physical and functional representations in the form of language are usually less adequate than literal or symbolic representations, but may be necessary when literal or symbolic representations are not available. Some learners will benefit if literal or symbolic representations are accompanied by language representations of the same phenomena.

Fidelity [low . . .high]

Fidelity refers to how close the representation resembles the real thing. For the physical view fidelity is the degree to which the representation resembles the actual entity, activity, or process. It implements the phrase "looks like." For the functional view fidelity is the degree to which the entity, activity or process performs similarity to the actual entity, activity, or process. It implements the phrase "acts like." Fidelity is represented by a continuum with values ranging from low to high. While there are some rules of thumb that would suggest that higher fidelity is always better, there are many situations where instruction is enhanced by lower degrees of fidelity.

Intelligence in Instructional Transaction Shells

Transaction shells incorporate intelligence about instructional design in several ways. First, the responsibilities and methods of each transaction shell enable the type of interactions most appropriate for acquiring a given type of knowledge. The designer does not need to reinvent appropriate instructional designs for every application. Second, the knowledge base includes a syntax[13] for knowledge representation that not only enables transactions to use this knowledge, but assures that the knowledge included is complete and consistent. Third, intelligence in the knowledge acquisition system enables subject matter experts to supply the necessary knowledge without knowing the formal syntax of the knowledge representation system. Fourth, the parameters of each transaction shell identify those ways that these interactions can vary for different learner populations and different tasks. Fifth, intelligence in the transaction configuration system contains instructional design rules relating learner and task attributes to various values on the instructional parameters. Thus, the instructional designer needs to supply only descriptive information about the learners and the task, the configuration system can select a pattern of instructional parameter values consistent with this information. However, the configuration system is merely a guide, a designer has access to all of the parameters and can adjust the value of groups of parameters or individual parameters to more adequately fine-tune a particular transaction instance.

[13] See Jones, Li, and Merrill (1990) for a description of this knowledge representation syntax and its relationship to other forms of computer-based knowledge representation.

Parameters

The transaction configuration system enables the designer to provide values for a wide range of instructional parameters. These parameters control the nature of the interactions with the learner. Instructional parameters enable a given transaction shell to be customized for a particular learner population, learning environment, and learning task. An instructional transaction shell is more than merely a computer algorithm that can use different subject matter data in different situations. It can be configured in many ways by changing its parameter values. Transaction shells can be modified to represent a complete range of instructional interactions.

Different transaction classes teach different kinds of knowledge and skill[14]. In this paper we will identify responsibilities and methods that characterize all instructional transactions regardless of the knowledge and skill that is taught. We will illustrate some of the parameters that are associated with these responsibilities and methods for a sample transaction class, but the detailed parameters associated with each class of transaction will be left for subsequent papers.

RESPONSIBILITIES OF INSTRUCTIONAL TRANSACTION SHELLS

The instructional design prescriptions of first generation instructional design are characterized as "best case" prescriptions. In our own previous work[15] we have identified different prescriptions for each of several performance-content outcomes. These prescriptions identify values for a number of variables which characterize the best instructional strategy for each of the possible outcomes. These prescriptions, however, do not identify the range of values for the many parameters which characterize each prescription, nor do they indicate conditions for which these parameters should assume different values. In other words, each outcome classification has a "best" case prescription and deviations from this prescription are left to the individual instructional designer. Transaction shells do not merely represent a best case. By changing its parameter values it can be configured in many ways to represent a complete range of instructional interactions.

Several instructional responsibilities are necessary in order for an instructional transaction to successfully interact with a learner. All instructional transactions, regardless of the type of knowledge or skill taught, must be capable of performing these responsibilities. The specific methods and parameters that are necessary for a given type of transaction to accomplish these responsibilities will differ from one class of transaction to another. In fact, the difference in the way these responsibilities are accomplished by different classes of transactions is one of the characteristics that distinguish one class of transaction from another.

Each responsibility is accomplished via several *methods* that are specific activities that enable the responsibility to be accomplished. These methods require values on a number of instructional *parameters*. These parameter values determine exactly how a given method is applied in a given transaction instance. The interactions enabled by a given transaction can exhibit a considerable variance depending on the values assigned to the parameters which constrain its methods. Instructional design via instructional transaction shells consists of selecting parameter values appropriate for a given learner population and particular learning task. These parameter values then enable the methods of each responsibility to carry out this responsibility in a way consistent with the requirements of a given learning situation.

---

[14] In a subsequent paper we will describe these transaction classes, the kinds of knowledge taught by each transaction class, and the relationships among these different transaction classes.

[15] Component Display Theory as described in Merrill, 1983; 1987; 1988.

Parameters

All instructional transactions must include the following responsibilities: *knowledge selection, knowledge sequence, interaction management,* and *interaction enactment.*

Select Knowledge

From all the knowledge associated with a given transaction instance, the knowledge selection responsibility determines that part which will be taught during a particular enactment of the transaction. Each of the frames in the knowledge base may include a large number of components: parts, steps, or events. Each of these knowledge frames may be implemented by several different mediations in the instructional resource data base. The amount of available knowledge often exceeds that which needs to be presented during a given enactment of the transaction. When a transaction is sent a message to do its job, the first parameters it needs are those which tell it of all the knowledge that is available, which specific knowledge elements are to be included during this enactment of the transaction.

Sequence Knowledge

The knowledge sequence responsibility determines which of the selected knowledge elements is presented next. Whenever the amount of knowledge to be included in a given enactment of a transaction exceeds that which should be presented[16] simultaneously, then an instructional transaction requires sequence parameters to indicate how this knowledge should be partitioned and sequenced. Knowledge acquisition is facilitated if the knowledge is partitioned into mind-size pieces; on the other hand knowledge assessment often requires the learner to interact with the knowledge as a whole. A given instructional transaction, regardless of the type of knowledge taught, should be able to invoke a variety of instructional sequences.

Manage Interactions

The instructional management responsibility determines how the student will interact with the selected and sequenced knowledge. Instructional management is accomplished by the selection of an *instructional strategy*. An instructional strategy is a sequence of *interaction modes,* each of which knows how to either overview information, present information, facilitate the students' practice of the skills promoted, or assess the students' knowledge and skill. The management responsibility also determines when a learner should move to the next interaction mode in the strategy. Instructional strategies can vary from providing information to promoting mastery of the knowledge and skills involved in the transaction. The type and sequence of interaction modes varies from one strategy to another. A given instructional transaction, regardless of the type of knowledge taught, should be able to invoke a variety of instructional strategies.

---

[16] The word *present or presented* is used in two ways in this paper. One meaning is to display information. However, in describing the knowledge involved in a transaction, our intent is that this is the knowledge with which the student interacts, hence the word *presented* in the context of an instructional transaction includes this notion of interaction, both display and response.

Enact Interactions

The instructional enactment responsibility determines how each interaction mode in a given strategy carries out its responsibility. The enactment responsibility determines the role a given interaction will play whether presenting information, enabling practice, or assessing a student. The enactment responsibility specifies how the interaction presents information, constrains learner responses, and/or reacts to the learners responses. The enactment responsibility also determines how each interaction is adjusted to provide the type of interaction most appropriate to a given student and subject matter. A given interaction mode, regardless of the type of knowledge taught, should be able to modify the nature of its interaction with the student in a variety of ways.

METHODS OF INSTRUCTIONAL TRANSACTION SHELLS

A transaction accomplishes each of its responsibilities via a set of methods. A method is a set of activities carried out by the responsibility in the process of enacting the transaction. The way that a particular method accomplishes its role is determined by a set of specific parameters.

All transactions include knowledge selection methods for *partitioning* knowledge, *portraying* knowledge, and *amplifying* knowledge; knowledge sequence methods for *routing* the learner through the selected knowledge and *guiding* the learners advancement; interaction management methods for *prioritizing* interactions and *expediting* the learners acquirement of the knowledge involved in the interaction; and interaction enactment methods for *overviewing* , *presenting* , *enabling practice,* and *assessing knowledge.*

Select Knowledge

The knowledge selection responsibility is accomplished via three methods: *knowledge partitioning*, *knowledge portrayal,* and *knowledge amplification*.

Partition knowledge

Learners can process only a limited amount of information at any one time. Working memory can effectively consider less than 7± 2 chunks of information simultaneously. When more information is presented previous information is lost from working memory. Effective instruction therefore requires that when the amount of information to be presented exceeds that which can be effectively handled by short term memory, then this information needs to be divided into mind-sized pieces. The *knowledge partitioning* method accomplishes this task by: first, selecting from the knowledge available to the transaction that knowledge which will be taught, and second, while making this selection dividing the knowledge into mind-sized pieces or segments.

Portray knowledge

A primary advantage of learning from an instructional situation is that the learner can be brought into contact with a much wider scope of knowledge via the mediation of this knowledge than would be possible if the learner was required to learn only from the real world or on-the-job. Not only can a wider scope of knowledge be presented but this knowledge can be mediated in ways that enable it to be more effectively learned than learning it from its naturally occurring state. The *knowledge portrayal* method selects from the various mediations available in the resource data base those knowledge mediations that will be used by the current enactment of the transaction.

Amplify knowledge

The knowledge to be taught can be divided into primary knowledge and ancillary knowledge. Primary knowledge is that information that is the focus of a particular instructional intervention. Ancillary knowledge is additional information that is prerequisite to the primary knowledge, that provides nice-to-know information related to the primary knowledge, that provides a context or setting for the primary knowledge, or which makes the primary knowledge easier to learn. The *amplify* knowledge method selects from the available ancillary knowledge that which will be included in a particular enactment of the transaction.

Sequence Knowledge

The knowledge sequencing responsibility is accomplished via two methods: *routing* the learner through the selected knowledge and *guiding* the learners advancement through the selected knowledge.

Route learner

A path orders the knowledge segments for a given student. The route learner method determines an appropriate path through the selected knowledge; that is, at any point in the instruction which knowledge segment a given learner should study next.

Guide advancement

The guide advancement method determines when a learner has received maximum benefit from study of a given knowledge segment, and when the student should move to the next knowledge segment. Under learner control this method also provides appropriate information to enable the leaner to choose from alternative paths.

Manage Interactions

The interaction management responsibility is accomplished via two methods: *interaction prioritization* and *acquirement expediation*.

Prioritize interactions

Each transaction consists of a number of possible *interaction modes*. An instructional strategy is an ordering of these interaction modes for a given student. This method determines which interaction mode should be engaged next for a given student.

Expedite acquirement

Acquirement refers to the learners internalization of the knowledge and skill being taught by a given transaction. Interaction modes are the means by which this acquirement is accomplished. The expedite acquirement method determines when the student has received maximum benefit from one interaction mode and should move to the next in order to optimize this acquirement. Under learner control this method also provides appropriate information to enable the learner to choose from alternative interaction modes.

Enact Interactions

The interaction enactment responsibility is accomplished via four methods corresponding to each of the primary interaction modes: *overview knowledge, present knowledge, enable practice, and assess knowledge.*

Overview knowledge

The overview method enables the learner to see the big picture, to see individual knowledge components in a larger context. Overview provides a quick look at the knowledge without extensive interaction.

Present knowledge

The present method is the primary way that knowledge is conveyed the learner. Present does not imply passive interaction; an adequate presentation may involve a great deal of interaction. However, the primary purpose of the interaction is to provide the knowledge to the learner.

Enable practice

The practice method enables the learner to consolidate their knowledge. Practice provides opportunity for the learner to demonstrate, generalize, elaborate, integrate, and extend new skills.

Assess knowledge

The assess method enables the system to determine the extent to which the learner has internalized the knowledge. To what extent is the learner able to remember the knowledge, use the knowledge, and apply the knowledge. Assessment is similar to practice but with most of the learning supports removed to enable the learner to demonstrate knowledge acquisition in situations which resemble real world settings.

PARAMETERS OF INSTRUCTIONAL TRANSACTION SHELLS

Each of the methods of a transaction shell are determined by parameters associated with the method. Different types of transaction shells have different parameters associated with their methods. One of the things that distinguishes one class of transaction shells from another is the way a particular method carries out its mission and the nature of the parameters which controls the actions performed by the method. In this section we will use the identify class of transaction shells as an illustration of parameters.

As knowledge an identify transaction requires either an instance or class entity frame. The formal knowledge required includes a structural representation identifying the names of the entities or parts; the clustering of the entities, parts and subparts; and the properties and their legal values which are associated with each entity or part in the knowledge structure. The primary resource data base required includes a physical representation on which the name, location, and function of each part can be identified, and/or a functional representation on which the name, location and function of each component can be identified. Secondary knowledge may include either textual or audio representations for the pronunciation, description, function, and notes (asides) associated with each entity or part in the knowledge structure.

An identify transaction enables the learner to acquire the names, functions, properties, and relative location of all the parts which comprise an entity. The learner knows *what* it is. Learning the names, properties, location, and function of the parts of a entity is a prerequisite to learning how an entity works, or how to operate an entity. Students are shown a physical representation of the entity and asked to identify individual parts, their function, and their immediate connections. Students are shown functional diagrams and asked to identify individual components, their function, and their immediate connections. Students are shown both the physical and functional representations and asked to demonstrate the correspondence between these two representations.

The identify transaction must both present all or a subset of the names to the learner and enable the learner to practice locating the parts and identifying the part name, function and properties. The transaction must present the functional names to the learner, must pair the functional names with their physical referents. The transaction must enable the learner to practice identifying functional symbols given referents; referents given the functional symbols; names of functional symbol given its graphic representation; and reproducing the symbols.

The parameters described in the following paragraphs are some of those which enable the methods of the identify transaction to carry out these interactions using the knowledge associated with this type of transaction. A particular instantiation of an identify transaction may include other parameters not identified here or may exclude some of the parameters identified here.

KNOWLEDGE SELECTION PARAMETERS

Knowledge selection methods and parameters determine from all the knowledge in the knowledge base and associated resource data base which could be taught by a given transaction shell which knowledge will be taught.

Selection control [LC, SC][17]

The first parameter associated with every transaction shell responsibility is that of control. Who makes the functional decisions, the system or the learner? Selection control determines who makes knowledge selection decisions: the learner or the system. A student can be given access to the complete knowledge base associated with a given transaction shell and allowed to select the material to be learned, or the knowledge can be selected by the instructional system as determined by the parameters of the partition, portray, and guide methods.

LC determines that the learner has access to the entire knowledge base associated with a given transaction and its associated resources. The learner can select what knowledge

[17] We have used the convention of square brackets to include the values that a given parameter can assume. A value which must be supplied by the designer or instructor is indicated by quotation marks. Values not in quotation marks are options which the designer can select.

components to study, which of the available resources to use, and which of the available ancillary information to examine. The value selected for this parameter becomes the default value for each of the control parameters associated with each of the selection methods. One option for the learner under LC is to select systems control. It is therefore necessary under conditions of both LC and SC for system control parameters values to be assigned.

Partition knowledge

A given knowledge frame may contain more information than is appropriate for a given interaction at a particular time in the instruction. An entity usually includes more parts than the learner can remember at one time. This knowledge must then be partitioned into "mind sized" knowledge segments for presentation.

For an entity each coordinate set of parts, those which share a superordinate entity or part, is considered a segment. A student will usually interact with all of the parts in a segment simultaneously during the enactment of the identify transaction. The maximum size of a given segment is limited by the knowledge base parameter of chunk size. The partition method for an identify transaction partitions the entity component knowledge base into segments consisting of the coordinate sets of parts.

In a given situation it may be unnecessary to teach all of these segments to the student. For certain situations it may be unnecessary to teach all of the parts in a given segment. Three knowledge partitioning parameters determine which segments are available for a given enactment of the identify transaction: *partition control, focus* and *levels.* A forth parameter, *coverage*, determines which parts will be included in each of the segments.

Partition control [SC, LC]

LC displays the entire knowledge base and allows the learner to select which segments and which components in these segments to examine. SC is determined by the focus, levels, and coverage parameters.

Focus ["name"]

Focus means the component of the knowledge structure where the instruction starts and which is the emphasis of the instruction. Focus can be the entire entity knowledge frame, or focus can be some component part within the knowledge frame. The user specifies the frame or the component part *name* which is the focus for a given segment.

Levels [all, "n"]

Levels determines how much of the knowledge cluster below the focus is included. *All* means that the entire cluster from the focus downward is included. When the focus is the entire knowledge frame then *all* means that every component identified in the knowledge base is included. When the focus is some component of the knowledge frame then *all* means all of the components that are subcomponents of the focus component. The parameter *n* is set by the user and indicates the number of levels of detail below the focus component that is included in the segment.

Coverage [all, "user"]

For each segment or chunk at each level coverage determines which components are included in the segment. *All* determines that the entire coordinate set or chunk is to be included; *user* means that the designer will specify which components from each coordinate set or chunk are to be included.

Figure 7 illustrates the partitioning method and parameters for a parts cluster diagram from an entity knowledge base. Because of the values selected for the partitioning parameters only three of the potential segments will be taught.

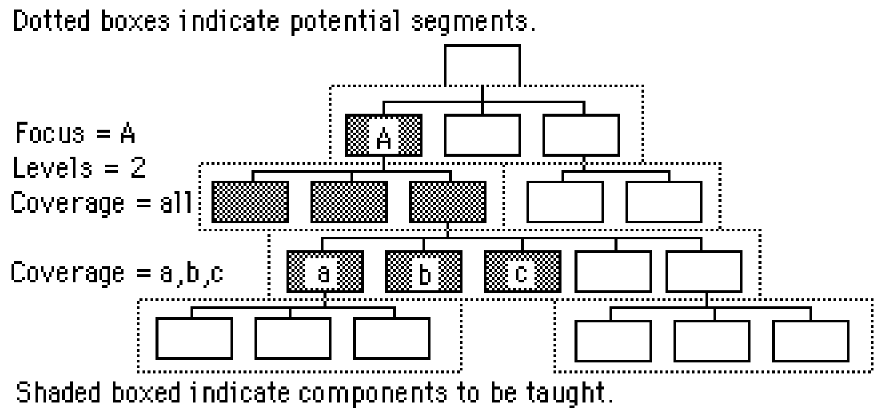


Figure 7 Partitioning an entity knowledge base

Portray knowledge

The portray knowledge method enables alternative resources, if they are available in the resource data base, to be used for the transaction. The selection of which resource to use for a given enactment of the transaction is determined by four parameters: *portrayal control, view, mode,* and *fidelity*. If a resource data base has only a single representation, or only one representation for each view, physical and functional, then one or more of these parameters are not relevant and the transaction shell uses the available resource(s).

Portrayal control [SC, LC]

LC provides a menu for view, mode, and fidelity and allows the learner to select the view to examine, the mode of representation for this view, and the level of fidelity. Selections which are not available in the resource data base do not appear on the learner control menu. SC is determined by the availability of alternative resources and the parameters of view, mode, and fidelity.

View [structural, physical, functional]

For a structural view the method constructs the structural tree diagram from the knowledge base and displays this tree diagram to the student. For physical or functional view the method displays the selected view from the resource data base.

Mode [language, symbolic, literal]

If alternative modes of representation exist in the resource data base then mode selects the view specified by view using the mode specified by mode.

Fidelity [low ... high]

If alternative levels of fidelity of representation exist in the resource data base than fidelity selects the view in the specified mode at the specified fidelity.

Amplify knowledge

The amplify knowledge method provides the learner with appropriate ancillary information providing that it is available in the resource data base. This ancillary information is usually presented as hypertext or audio which overlays the primary instruction which is underway as part of the transaction. A number of parameters control the presentation of this information.

Ancillary information control [SC, LC]

This control parameter sets the control for all ancillary information, or each type of information may be either LC or SC. For LC the user has a menu or other signal to indicate that ancillary information is available by selecting the signal or menu item. Menu items or signals do not occur when ancillary information of particular type is unavailable. For SC the ancillary information is presented as part of the regular instruction.

Ancillary information mode [verbal, audio]

All of the ancillary information may be in the same mode as an overlay text window or as overlay audio, or the mode for each type of ancillary information may be determined individually. When selected verbal information is shown in a overlay text window, audio information is signalled by a icon on the screen and then played for the learner.

Pronunciation availability [no, SC, LC]

A *no* parameter value means that the information is available in the resource data base but should not be presented to the learner during the enactment of the transaction. This is important during practice or assessment when ancillary information may provide undesired help to the learner and interfere with internalization or assessment of the knowledge.

Pronunciation mode [verbal, audio]

Determines mode for available pronunciation information.

Component function availability [no, SC, LC]

Determines if function information which is available in the resource data base is to be available to the student and whether the student or the system determines if and when to display this information.

Component function mode [verbal, audio]

Determines mode for available function information.

Component description availability [no, SC, LC]

Determines if description information which is available in the resource data base is to be available to the student and whether the student or the system determines if and when to display this information.

Component description mode [verbal, audio]

Determines mode for available description information.

Component aside availability [no, SC, LC]

Determines if aside information which is available in the resource data base is to be available to the student and whether the student or the system determines if and when to display this information.

Component aside mode [verbal, audio]

Determines mode for available aside information.

KNOWLEDGE SEQUENCE PARAMETERS

Knowledge sequence methods and parameters determine the order in which the knowledge segments will be presented to the student. The entire sequence determination can be given to the learner or it can be controlled by the system.

Sequence control [LC, SC]

LC determines that the learner has access to the segments selected by the selection responsibility and can determine what segment to study next and when to move to this segment. The value selected for this parameter becomes the default value for each of the control parameters associated with each of the sequence methods. One option for the learner under LC is to select systems control. It is therefore necessary under conditions of both LC and SC for system control parameters values to be assigned.

Route learner

Knowledge frame components -- parts for entities-- usually involve subparts for several levels of detail. For an entity knowledge frame this means that the entire entity is first divided into 7 (± n) parts or subassemblies. Each of these subassemblies is then divided into 7 (± n) parts or subassemblies until all of the parts of the entity are identified. Components high in the cluster represent very large sections of the entity while those at the lowest levels represent the most detailed parts appropriate to the nature of the subject matter. Route learner parameters determine the way for the learner to traverse the knowledge segments which partition these levels of detail for a given enactment of a transaction.

Segment sequence control [LC, SC]

Sequence control determines who makes segment sequence decisions, the learner or the system. One option under learner control is the option to select system control. Therefore, values must be supplied for system control parameters even when learner control is selected. *LC* determines that the learner can control direction through the segments of the selected knowledge. *SC* determines that the system will direct the traverse of the segments directed by the parameters sequence type, depth, direction, and accrual.

Segment sequence type [elaboration[18], cumulation, accrual, "user"]

Sequence type refers to whether the learner moves from the top of the component structure down, from the bottom of the component structure up, horizontally across the most detailed components of the structure, or in a direction determined by the user. Top down is called *elaboration*.. The learner is provided the top level segment or the "big picture." Each component of the big picture, top level segment, is then elaborated to show its detail. Bottom up is called *cumulation*. The learner is presented segments of the most detailed components first. These detailed segments are then combined into more complex parts in segments at the next higher level. *Accrual* means that the student learns only detailed components. The level of detail is the terminal level of detail as determined by the levels parameter. If the levels parameter is *all*, then this is the most detailed level of components available in the knowledge base; if the levels parameter is *n* then the accrual is at the lowest level of detail specified by *n*. The *user* value means that the designer can indicate any sequence of the segments in the selected knowledge.

Depth [depth first, breadth first]

If segment sequence is elaboration or cumulation then the depth parameter modifies the direction. For elaboration *depth first* means that all of the subcomponents to the most detailed level are presented before presenting the subcomponents of the next component. For cumulation, *depth first* means that all of the specific components are cumulated into the component at the highest possible level before going on to the next chunk of detailed components. For elaboration *breadth first* means that all of the segments of components at one level are taught before moving to the next lower level. For cumulation *breadth first* means that all of the most detailed segments are taught before moving to the next higher level to cumulate these detailed components into more general components.

Accrual [all, isolated part, replacement]

If segment sequence is accrual then the accrual parameter indicates the sequence for presenting the most specific components. *All* determines that the entire set or sequence of detailed components is presented simultaneously. *Isolated part* means that a chunk of items is presented. After the transaction has determined that the learner has completed this chunk of components the next chunk is presented. *Replacement* is a value appropriate for entity knowledge frames and means that the new segment includes items that have previously been learned in previous segments. In other words specific components are represented in subsequent segments. Higher levels of components are not presented as part of an accrual sequence.

---

[18] An elaboration sequence is similar to the prescriptions of Elaboration Theory (see Reigeluth & Stein, 1983; Reigeluth, 1987).

Priority [chronological, frequency, critically, familiarity]

If accrual is isolated part, then priority refers to the order of introduction of the segments of components. For entities chronological sequence is often arbitrary and is defined from left to right as the parts are represented by the subject matter expert in the component structure. *Frequency* means that the most frequently used parts are presented first with less frequently used components being introduced as the instruction progresses. Frequency requires the subject matter expert to include frequency ratings with the structural representation of the components. *Criticality* means that the most critical parts are presented first with less critical components being introduced as the instruction progresses. Criticality requires the subject matter expert to include criticality ratings with the structural representation of the components. *Familiarity* means that the most familiar parts are presented first with less familiar components being introduced as the instruction progresses. Familiarity requires the subject matter expert or learner to include familiarity ratings with the structural representation of the components.

Guide advancement

The guide advancement method determines when a learner should move to the next segment in the selected knowledge. The information available to make this determination is limited by the instructional strategy selected for the instructional management method. The guide advancement method involves a degree of on-line monitoring and advisement. Three parameters determine how the guide advancement method carries out its duties: *shift segment -on, repetitions,* and *criterion.*

Shift segment -on [LC, repetitions, practice criterion, assessment criterion]

The shift segment -on parameter determines the data used to recommend a shift to the next knowledge segment. LC means that the learner can determine when to go the next knowledge segment. Usually this learner control is advised, that is, the learner is told which components have not been examined or mastered and ask to confirm the decision to go to the next segment. Repetitions affects the presentation method and means that the decision to move to the next segment is made after the student has been shown each component at least "r" times. It is consistent with a familiarity, basic, mastery, or remedial interaction strategy. Practice criterion affects the practice method and means that the decision to move to the next segment is made after the student has gotten each component correct in practice "c" times. It is consistent with a basic, mastery, or remedial interaction strategy. Assessment criterion affects the assessment method and means that the decision to move to the next segment is made after the student has gotten each component correct in assessment "c" times. It is consistent with a mastery, or remedial-mastery interaction strategy. Within the limitations of a given interaction strategy it is possible to "AND" any combination of these shift-on values meaning that the presentation is given for "r" repetitions AND/OR the learner must correctly identify "c" practice AND/OR assessment components before shifting to the next segment. If LC is also included in the AND then the learner can decide when to shift to the next segment after the repetitions "r" AND/OR criterion "c" have been met.

Repetitions ["r"]

The number of times each component and its selected ancillary information is presented to the student. The value of "r" is usually an integer less than 5.

Criterion ["c"]

The number of times each component and its selected ancillary information is correctly identified by the learner. There may be a separate value of "c" for each type of ancillary information if desired. The value of "c" is usually an integer less than 5. Criterion implies that during practice or assessment the items will be sampled with replacement, that is, each time a learner misses an item during practice this item is presented for student identification again until the student has reached the criterion.

INTERACTION MANAGEMENT PARAMETERS

Interaction management methods and parameters determine which interaction modes will be used by the transaction , the order in which these interaction modes will be enacted, and when the learner should end interaction with one mode and shift to the next interaction mode.

Management control [LC, SC]

Interaction management LC determines that the learner has access to the entire set of interaction modes and can select interaction with any mode at any time. The learner can also determine when to shift to the next interaction mode or when to return to a previous interaction mode. SC is determined by methods and parameters for prioritizing interactions and expediting acquirement by determining when the student should shift to new interaction mode or return to a previous interaction mode. Since an option of LC is for the student to request SC these parameters values must be specified in either case.

Prioritize interactions

An interaction strategy is a sequence of interaction modes thought to be appropriate for a given level of learning involvement. Overview and familiarity strategies provide a quick look at the information to be learned and are useful for providing information but not insuring that this information has been internalized. Basic and mastery strategies involve the learner in a more intense learning experience involving practice and for mastery assessment to ascertain the degree to which the learner has acquired the knowledge being taught. Remediation strategies are appropriate when learners have already learned part or all of the information and need to have this knowledge refreshed or extended.

Strategy control [LC, SC]

Under LC the learner is provided a menu of the available interaction strategies and is able to make their own selection as to which strategy is most appropriate for their learning goals. Under SC the designer makes this selection based on information external to the learner as to the level of learning desirable.

Interaction strategy type [overview, familiarity, basic, mastery, basic-remediation, mastery-remediation, "user"]

An overview strategy provides the learner with the overview interaction mode. A familiarity strategy provides the learner with an overview plus a presentation modes. A basic instruction strategy provides the learner with overview plus presentation plus practice. Mastery adds assessment to the basic strategy. Remediation strategies assess the student and then provide either a basic or mastery strategy for those knowledge components for which the learner is unable to meet the established criterion. Selecting the user value enables the designer to determine some unique combination of interaction modes thought to be appropriate for a given subject matter or population of learners. (See Table 2.)

Expedite acquirement

The expedite acquirement method and parameters determine when a learner has attained maximum benefit from a given interaction mode and should move to the next interaction. Or when a learner should return to a previously used interaction mode. The expedite acquirement method involves a degree of on-line monitoring and advisement to assist the student in acquiring the desired knowledge as efficiently and effectively as possible.

Shift interaction - on [LC, repetitions, criterion, response time, elapsed time][19]

The shift interaction -on parameter determines the data used to recommend a shift to a next or a previous interaction mode. A value on this parameter is set for each of the interaction modes in a particular strategy.

LC means that the learner can determine when to go the next interaction mode or return to a previous interaction mode. Usually this learner control is advised. When the learner elects to shift they are told which components have not been examined, examined less than "r" repetitions, or which components have not been mastered to level "c". Given this advisory information they are ask to confirm the decision to go to the next interaction. Or the advisor function may determine that the learner has spent enough time, has attained the set repetition level "r", or the criterion level "c" in a given interaction. The advisor then interupts the learner to advise that the learner may want to move on. Under learner control the learner can override this advice.

The various parameters "r", "c", "t", and "e" can be combined by logical operators AND or OR. Under the AND condition both a certain number of repetitions must be completed and a certain criterion reached and a certain response time reached before the student is advised or allowed to move to the next interaction. Or when all of these conditions have been met the learner is advised or required to shift to the next interaction. Under an OR condition the leaner is advised or allowed to move to the next interaction when either "r" or "c" has been attained. Or advised or required to remain until either "r" or "c" has been attained.

Repetition ["r"]

Repetitions means that each part or component has been seen at least "r" times. The learner can be advised or required to shift after "r" has been attained, or advised or required to remain in a given interaction mode until "r" has been attained.

---

[19] These values are suggestive only. More elaborate advisor schemes can be devised involving learner aptitude variables which interact with performance variables. These more elaborate schemes will be left for subsequent work on advisor systems.

Criterion ["c"]

Criterion means that each part and its selected ancillary information has been correctly identified by the learner "c" times during either practice or assessment. The learner can be advised or required to shift after "c" has been attained, or advised or required to remain in a given interaction mode until "c" has been attained.

Response time ["t"]

Response time is the time that for responding to a given item in practice and assessment and is appropriate only when quick response to the identification of a part is a critical learning goal. The learner can be advised or required to shift after "t" has been attained for each component, or advised or required to remain in a given interaction mode until "t" has been attained for each component.

Elapsed time ["e"]

Elapsed time is the amount of clock time spend in a given interaction mode. A student may be advised or required to shift after "e" has elapsed. Or a student may be advised or required to remain in a given interaction until "e" has elapsed.

INTERACTION ENACTMENT PARAMETERS

The interaction enactment responsibility determines how each interaction mode in a given strategy carries out its duties. The parameters for a given interaction mode may be the same for all occurrences of this interaction mode, or the parameters for each occurrence of a given interaction mode may be different as required by a particular instructional situation.

A particular implementation of a certain class of transaction will result in a number of unique parameters occasioned by the particular implementation. In this paper there is not room to provide a complete specification for an identify transaction. Consequently the following methods and associated parameters are suggestive and provide an incomplete specification for an identify shell. These parameters are included to indicate the type of parameters that these methods require.

Enactment control [LC, SC]

All of the interaction modes can have the same control value set by this parameter or each of the interaction modes can have their own value of control set by their individual control parameters. Under LC the learner is provided the alternatives previously identified in Table 1. Under SC the system selects the alternatives identified in Table 1.

Overview Knowledge

The overview interaction mode provides the learner with the structure of the knowledge. For an identify transaction this is usually represented by a parts cluster diagram such as that represented in figure 7. In some situations allowing the student to browse this structure may be sufficient but in other situations it is necessary for the learner to correlate this clustering of parts with their appearance in either the functional and/or physical representation of the entity. The parameters of the overview method determine these options.

Overview control [LC, SC]

Sets the control level for the overview mode only as per the alternatives in Table 1.

Overview view [structure, + focus, + level 1]

The view parameter determines what is presented to the learner as part of the overview. Structure presents only the part cluster in the format determined by the structure format parameters. The +focus value presents the structure plus either the functional or physical representation of the focus segment with the focus component highlighted. The +level 1 value presents the structure plus either the functional or physical representation of the focus and the representation of the components of the focus. In this situation there would be three windows on the screen one showing the parts cluster, one showing the focus part in context, and the third showing the details of the parts of the focus component. The implementation[20] of this presentation would enable the student to traverse down the cluster diagram to any appropriate level with the corresponding resources changing as the focus of the learner is changed.

Structure format [tree, browser]

The structure format parameter determines how the parts cluster information is displayed to the student. The tree value provides a tree diagram showing the parts to at least three levels of the tree in one view with the provision for the learner to traverse up and down this tree as required. Browser uses a series of windows in which the focus component name is shown in one window, an adjacent window shows the parts of this focus, and a third adjacent window shows the parts of a selected part in the second window. Clicking on a part in window two provides a list of the parts of this part in window three. There are other formats that a given implementation may utilize and which would be included in a format parameter.

Present Knowledge

Presentation display methods and parameters determine details for each type of presentation. Particular display parameters differ from one type of presentation to another within a given transaction class, and among presentation types appropriate for different transaction classes. It is beyond the scope of this paper to provide a complete list of the possible presentation display parameters for the identify transaction class and each of the variations of presentations within this transaction class.

The following provides a sample of display parameters for the identify transaction class. The presentation of an identify transaction presents the location, labels, function, properties, pronunciation of each part of some entity. The elements of this interaction are *the location, label, function*., and *properties*. Other implementations may include other elements or exclude some of these elements. Display parameters determine which of these elements are presented, in what sequence, the timing for these elements, and who makes the decision.

[20] The implementation details of each of these interaction modes requires a more detailed specification of the transaction shell which is beyond the scope of the present paper.

Presentation display element control [LC, SC]

Under LC the leaner can select a component and determine whether to see the label, the function, and/or the properties for each part of the entity; the order in which this information is presented; and when to go the next part. The learner may also select SC as an option requiring the other parameters of the presentation method.

Presentation display element availability [label, function, properties]

If display element control is SC, then *label* determines that only the label will be displayed; *function* determines that only the function will be displayed; and *properties* determines that only the properties will be displayed. These values can be ANDed meaning that the label AND function AND properties can be displayed.

Presentation display element timing [untimed, n seconds]

If display element control is SC, then there is a time parameter associated with each of the elements to be presented. *Untimed* means that an element is displayed until the learner requests the next element. A value of *n* indicates that the element will be displayed for n seconds before the next element is displayed. A separate display element timing value must be associated with each element (label, function, and properties).

Presentation display element sequence ["order", simultaneous, sequential]

If two or more component elements are to be presented then there are a number of sequencing possibilities. All three elements can be presented simultaneously; or all three elements could be presented for each part, but one by one; or all of one element could be presented for each part , on a a second pass through the components all of the next element presented, etc. In either case the order of the elements may vary. For three elements there are 6 possible orders -- LFP, LPF, FLP, FPL, PLF, PFL.

Enable practice

The practice interactions for each transaction class differ from one another due to the nature of the knowledge to be practiced. The enable practice method and parameters determines the nature of this practice. The appropriate types of practice for a given interaction is determined by the transaction class involved. Within a given transaction class there may also be a variety ways for the learner to interact with the knowledge, different types of practice. In this paper we have not attempted to identify the all of the different types of practice which may be appropriate for an identify transaction.

The following provides a sample of practice parameters for the identify transaction class. Practice for an identify transaction requires the learner to locate and supply the labels, functions, and/or properties of each part of some entity. The elements of this interaction are the *location*, *label*, *function*., and *properties*. Practice parameters determine which of these elements constitute the display and which the response, the sequence of different practice formats, the number of repetitions for each part of the entity, the response mode, and response timing.

Practice formats ["list']

In an identify shell practice consists of the system providing part of the information associated with a given component and having the learner provide additional information. For example, the system could provide a label and have the learner locate the component. The parameter consists of a list of formats in the order that they are to be applied. These formats are specified by the following matrix. The entries indicate what information is provided by the system, S, and what information the learner is expected to supply, L. If the "list" contains more than a single format identification then a given practice can use more than one format either simultaneously or sequentially (see practice format sequence). Different instantiations of an identify transaction may include other information elements than those indicated.

Table 3 Practice formats for an idenfity shell

Format	Locate	Label	Function	Properties
1	S	L	L	L
2	S	S	L	L
3	S	S	S	L
4	L	S	S	S
5	L	L	S	S
6	L	L	L	S

Practice format sequence [sequential, simultaneous]

If there is more than 1 practice format in "list" then the format sequence parameter indicates when these different formats will be used. Sequential means that the first format in the list will be used on the first pass through the components, followed by the second format, etc. Simultaneous means that the format used for a given component will be selected randomly from the list by the system and will vary from component to component.

Response mode [recall, recognize]

If the learner must provide the label, function, or properties then a response mode of *recall* determines that the learner will type the label, function or properties; *recognize* determines that the learner will select the correct label, function or properties from a list.

Response timing [untimed, "n-seconds"]

An *untimed* response determines that the learner can take as long as necessary to indicate their response. A timed response determines that the learner must respond within the time limit set by *n-seconds* or a time-is-up message will be displayed and the learners response scored as incorrect. A timing parameter is selected with each element of the format that the learner will be required to provide.

Practice format control [LC, SC]

If more than one practice format is selected, then under LC the leaner can determine the type of practice format that they wish to use and the sequence of the these practice formats.

Response repetition ["n", contingent]

Under a practice format control value of SC the response repetition value "*n* " indicates how many times the learner is required to respond to each part. Response repetition acts in cooperation with practice format. If response repetition is set to 1 and practice format sequence is set to sequential, then only the first format will be used with each part. If response repetition is set to 1 and format sequence is set to simultaneous, then the format for each part will be selected at random from the "list" of formats. A value of *contingent* determines that under SC repetition occurs when the learner fails to meet the response criterion "c".

Component order [LC, same, random]

When a given set of parts and associated elements is presented or practiced more than once, component order refers to whether these components will be presented in the same order each time, a different order each time, or selected by the learner.LC determines that the learner can view the items in any order. *Same* means that the items are always presented in the same order; *random* means that the item order is randomly determined for each subsequent presentation.

Feedback availability [yes, no]

Feedback applies to practice and assessment interaction modes. Many tasks have intrinsic feedback, that is in the act of performing the task the consequence indicates to the learner whether or not the intended outcome occurs thus, obtaining confirmation as to the adequacy of the performance. In some instructional tasks, however, such intrinsic feedback is not available or not apparent to the learner. In these tasks extrinsic feedback must be provided to indicate the adequacy of the learners response. While convenient rules of thumb suggest that feedback is always desirable, there are occasions where the learning that occurs is hindered by too much feedback at the wrong time. Five parameters are associated with feedback: *availability, type, control, timing, and schedule.* Availability determines whether or not feedback should be presented for a particular response.

Feedback type [intrinsic, correct answer, right-wrong, attention focusing, "user"]

If feedback is available, then type determines the nature of this feedback. *Intrinsic* feedback is available as a consequence of performing the task. *Correct answer* provides the learner with the correct answer; *right-wrong* indicates whether or not the learner is right or wrong, but does not necessarily provide the correct answer if the learner is in error; *attention focusing* indicates that the feedback directs the learner's attention to the relevant aspects of the task in the hope that the learner will then be able to provide an appropriate response. "User" indicates that the designer can specify a unique type of feedback not included in the above parameter values.

Feedback control [LC, SC]

Control determines who decides when feedback will be presented and what type of feedback will be presented, the system (SC) or the learner (LC). Under *LC* the learner can request a particular type of feedback at any time following a response.

Feedback timing [immediate, schedule, delayed]

If control is SC, then feedback timing determines whether feedback will be presented after every response (*immediate*), will presented on some *schedule* (see feedback schedule parameter), or will be delayed after a complete execution of some activity or solving of some problem associated with a process (*delayed*).

Feedback schedule type [fixed interval, variable interval, fixed ratio, and variable ratio]

If feedback timing is schedule, then schedule type determines the type of schedule that will be used. In *fixed interval* the feedback occurs after a fixed period of time, whereas in *variable interval* the feedback occurs after a random amount of time within some range which changes for each feedback occurrence. In *fixed ratio* feedback occurs after a fixed number of responses, whereas in *variable ratio* feedback occurs after a random number of responses within some range which changes for each feedback occurrence. If feedback timing is selected, then a schedule must also be selected.

Assess knowledge

Assess knowledge is similar to practice and requires similar or identical parameters especially in the case of an identification transaction. We have not identified additional assessment parameters in this paper.

CONCLUSION

To the reader an immediate concern arises. One goal of Instructional Transaction Theory is to make instructional development more efficient -- an order of magnitude more efficient (See Merrill, Li, & Jones, 1990). But if even a simple instructional transaction -- learning the names of the parts of some entity -- requires the large number of parameters identified in this paper, then doesn't concern with all of these parameters make the design more time consuming rather than less? The answer is yes and no. Yes, if a designer had to individually set all of these parameter values in order to design a piece of instruction, or worse, had to concern themselves with each of these variables in designing unique instruction, then the time would increase. Paradoxically, hand wired instructional design must be concerned with all of these parameters anyway, but we do so implicitly rather than explicitly. No, if a piece of computer code captures how to teach identification and all of these parameters are part of the code but not made explicitly available to the designer then the instructional design can be significantly more efficient. A given designer would, via the transaction configuration system, provide information about students and the task. A set of rules --intelligence-- in the transaction configuration expert system would then convert this readily available information into a set of default values for the parameters of the transaction shell. The designer can then immediately inspect an enactment of the instruction and modify it by either changing the original input information about students and the task or, if necessary, explicitly changing a given parameter value effecting the part of the instruction they want to modify. It is analogous to using many of the modern computer interfaces. Programing in Windows or on the Macintosh is far more complex than earlier systems, but the resulting user interface, for the non programmer, is much easier to use. In a similar way, the level of detail suggested by this paper is necessary in order to program an instructional transaction shell, but, once in place, the subject matter expert user will be able to design and development technology based instruction much more easily.

REFERENCES

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

Li, Z., & Merrill, M. D. (1990). Transaction shells: a new approach to courseware authoring. Journal of Research on Computing in Education. 23(1), 72-86.

Merrill, M. D. (1983) Chapter 9 Component Display Theory. In C.M. Reigeluth (Ed). Instructional Design Theories and Models: An Overview of their current status. Hillsdale, NJ: Lawrence Erlbaum Associates.

Merrill, M. D. (1987) Chapter 7 A lesson based on Component Display Theory. In C. M. Reigeluth (Ed). Instructional Design Theories in Action. Hillsdale, NJ: Lawrence Erlbaum Associates.

Merrill, M. D. (1988) Chapter 3 Applying Component Display Theory to the design of courseware. In D. H. Jonassen (Ed) Instructional Designs for Microcomputer Courseware. Hillsdale, NJ: Lawrence Erlbaum Associates.

Merrill, M. D., Li, Z. & Jones, M.K. (1990). Second generation instructional design. *Educational Technology,* 30(2), 7-14.

Merrill, M. D., Li, Z. & Jones, M.K. (1991). Instructional Transaction Theory: an introduction. *Educational Technology*. 31(6), 7-12.

Reigeluth, C. M. & Stein, F. S. (1983). The Elaboration Theory of Instruction. in C. M. Reigeluth (Ed) Instructional Design Theories and Models. Hillsdale, NJ: Lawrence Erlbaum Associates.

Reigeluth, C. M. (1987). Lesson blueprints based on the Elaboration Theory of Instruction. in C. M. Reigeluth (Ed) Instructional Theories in Action. Hillsdale, NJ: Lawrence Erlbaum Associates.