2008-06-01

The Architecture of Instructional Theory

Andrew S. Gibbons
andy_gibbons@byu.edu

Clint P. Rogers

Chapter 14

The Architecture of Instructional Theory

Andrew S. Gibbons

P. Clint Rogers

Dr. Andrew S. Gibbons
Chair, Department of Instructional Psychology and Technology
Brigham Young University
andy_gibbons@byu.edu

Dr. Andy Gibbons is department chair in Instructional Psychology and Technology at Brigham Young University. Prior to that, he was a faculty member at Utah State University. He led instructional design projects in industry for 18 years at Wicat Systems, Inc. and Courseware Incorporated. Dr. Gibbons' work has included large-scale training development projects, re-engineering of the development (ISD) process, computer-based instruction, military and commercial aviation training development, and research and development on instructional simulations. Dr. Gibbons' current research focuses on the architecture of instructional designs. He has published a design theory of Model-Centered Instruction, proposed a general Layering Theory of instructional designs, and is currently studying the use of design languages in relation to design layers as a means of creating instructional systems that are adaptive, generative, and scalable.

To be published in C. M. Reigeluth & A. Carr-Chellman, *Instructional-Design Theories and Models, Volume III*, Summer 2008.

P. Clint Rogers
Mqrriott School of Business
Brigham Young University
clint.rogers2008@gmail.com

Dr,. Clint Rogers teaches, conducts research, and does consulting work primarily regarding IDT, web analytics, and cross-cultural online collaboration and innovation. He has worked with Brigham Young University, and with the University of Joensuu, Finland, coordinating the Cross-Cultural Research Group and supervising dissertations in the IMPDET program (International Multidisciplinary PhD Studies in Educational Technology www.impdet.org). He holds a doctorate in Instructional Psychology and Technology, and has specific research interests in IDT, global virtual teams, fostering human potential, the cultural customization of online collaboration and innovation, the philosophical roots of education and science, and the impact of technological diffusion on international development, business, and social change - and obviously a key interest in layers and languages in instructional design. For his most recent work and interests, visit: www.clintrogersonline.com/blog.

Editors' Foreword

Vision

- To relate instructional theory and instructional design theory

Kinds of Theory

- Two Kinds of Theory: Instructional and instructional-design
- Design Theory: Applies across all domains of design
- Domain Theory: Is particular to a domain of design, e.g. instruction
- A basis for the design theory/domain theory distinction: multiple categories of engineering design knowledge
- Design instrumentalities and instructional design theory (functional decomposition versus process decomposition)

Design Layering by Functional Decomposition

- Employed in numerous design fields, including architecture, computer and software design, multimedia design, and others
- Being aware of layers allows us to design for dynamic and changing contexts.

Design Layering and ID

- The layering notion for ID includes:
  - Content layer
  - Strategy layer
  - Message layer
  - Control layer
  - Representation layer
  - Media-logic layer
  - Data management layer

Design Languages

- Design languages and natural languages differ in primitive terms, syntax, and semantics.
- A design language is abstracted through patterns from previous designs.
- As design languages evolve and we become fluent in using them, the result is advances in design sophistication, effectiveness, productivity, and quality of designs.

Operational principles and instructional theory

- Operational principles link design layers and design languages to instructional theory.

Layers, languages, operational principles, and instructional theory

- ID theory provides a structural framework of layers within which instructional theories can be analyzed and compared.
- There is a great deal of work in instructional theory that is related to layers.

— CMR & ACC

The Architecture of Instructional Theory

This chapter joins a discussion of instructional theory that has been ongoing for nearly a century. It departs in some ways from prior discussions: (1) it considers instructional theory as a species of technological theory rather than as a type of scientific theory,* a view expressed more fully elsewhere (Gibbons, 2003a), (2) it adopts the viewpoint articulated in earlier chapters of this book that there are multiple distinct bodies of technological theory that pertain to the work of instructional designers, (3) it attempts to articulate a particular view of the nature of two of those bodies of theory by describing their relationship to each other, and (4) it suggests a direction for the future exploration of additional bodies of theory, based on the writing of Vincenti (1990). Other views of possible theory

---

* Editors' note: This distinction is similar to the one made in Chapter 1 between design theory and descriptive theory.

development are described in Reigeluth and Carr-Chellman (Chapter 1) and Bichelmeyer, Boling and Gibbons (2006). Each of these views of the future development of theory begins from a different starting point and suggests interesting alternatives for exploration, perhaps leading to a new level of discussion of the role of theory in instructional design.

Most current practitioners of instructional design find it hard to describe in other than very general terms how instructional theories influence their designs. We feel this situation will improve if design theorists can provide a more nuanced view of instructional theory that relates theory more directly to everyday design concepts and practices. In this chapter we describe an architecture of instructional theory that ties the elements of an instructional design in a more detailed way to instructional theory. Rather than tracing the origins of a design back to a single instructional theory, this architecture suggests that different features of a design should be related to different, local, instructional theories. We propose that those local theories work within a larger framework of instructional design theory. These two different bodies of theory—instructional theory, and instructional design theory—and their relation to each other are the subject of this chapter.

Distinguishing Two Kinds of Theory

Our discussion highlights a distinction between *instructional theory* and *instructional design theory*, consistent with the discussion by Reigeluth and Carr-

Chellman (Chapter 1 of this volume) of *instructional-event* theory and *instructional-planning* theory. However, in this chapter we will adhere to the more familiar terms (instructional theory and instructional design theory) when referring to these two bodies of theory: first, in hopes of clarifying terms already in common use that have been the source of some confusion, and secondly, to maintain consistency with Vincenti's view of design.*

To begin, we should describe the contrast we see. In our view *instructional theory* deals with the structure of instructional conversations, and *instructional design theory* deals with the manner in which the elements of those conversational structures are selected, given dimension, and integrated into a design. This suggests that one body of theory (instructional design theory) provides a framework within which the second body of theory (instructional theory) can be applied. In this perspective, the substance of an instructional theory consists of categories of design building blocks and the rules by which building blocks may be articulated to form different designs. The substance of instructional design theory, on the other hand, consists of methods for analyzing and decomposing design problems, classes of design structure, and principles for deriving design processes appropriate to different types of design problems. If

* Editors' note: Our view is that the confusion over the term "instructional-design theory" is so long-held and deeply engrained that consensus will not be reached on a single meaning for it, and that we are better off to use a different term that is unambiguous. We agree that this distinction is very important; we only disagree on the terms that should be used.

instructional theory reflects a particular theorist's view of effective *instructional* structures and operations during instruction, then instructional design theory reflects a view of effective *design* structures and operations during designing.

This distinction between two types of theory related to instructional design parallels similar views of theory in design fields in general. In virtually all mature design fields there exist multiple domain theories that describe different theorists' views about fundamental building blocks and rules for articulating these building blocks together in workable ways. There exist at the same time in those fields theories that govern the *making* of designs. Both kinds of theory are critical to advances in design practice in those fields. From this point of view we give a more detailed account of instructional theory and its architecture by describing it within a framework of instructional design theory that is expressed in terms of *design layers* and *design languages*. We show how this view of instructional design theory makes possible a more detailed discussion of existing instructional theories and their comparison against a common background. A brief summary of our argument is given below, followed in later sections by a more detailed discussion of key points.

In the past, the most common approach to instructional design theory has been of generic design processes (primarily ADDIE), but we propose that process is only one of many possible approaches to the decomposition of design problems into solvable sub-problems. We consider an alternative decomposition scheme that has been used successfully in other design fields—decomposition in terms of artifact functionality—for example, the formation of message structures, the representation of message structures to the senses, and carrying out strategic interactions. Functional design decomposition creates separate design *layers* representing design sub-problems that can be addressed somewhat independently. Each layer accounts for the design decisions related to the individual functions that become integrated into a complete design. Design languages, which are collections of abstract structures, supply specific structures and qualities to designed features within these layers. Terms of design languages are supplied by the shared community languages of designers, which include among other things the vocabularies of specific domain theories: instructional theories. Problems within each design layer are solved using layer-related languages. Thus, every design is expressed in the terms of multiple design languages, each having a mixture of theoretical and practical bases.

The specific layers and sub-layers involved in a particular design (and therefore the languages used to create the design) evolve and change based on design decisions, constraints, criteria, resources, tools, new technologies, construction (development) methods, and available designer skills and awareness. For instance, commitment to a specific delivery medium (such as videotape) injects certain sub-layers and design languages into the design and may remove others (such as those for computers) from consideration. Therefore, each design includes its own unique combination of sub-layers. At the most detailed level, layers are created or destroyed according to the decisions and dynamics of a specific project. In this chapter we describe a list of high-level layers that we feel are generic to virtually all instructional designs.

In this view of instructional design theory, an instructional theory can be described as a domain theory — a set of specialized, mutually-consistent design languages that consist of defined terms distributed across multiple design layers. That is, an instructional theorist supplies building-block elements that constitute legitimate terms of designing for use within one or more layers. This insight describes the relationship between instructional design theory and instructional theory. Design theory provides the structural framework within which specific instructional theories can be analyzed and compared. Instructional theories work within a framework of functional design layers, however those layers are construed by the theorist.

Design Theory

As we have said, design theory is a body of theory about design making that can be considered independently of the specific fields in which the designs are made. Simon (1999) describes how attention to design architectures, design processes, and design theory have been forced on us by the introduction and widespread use of the computer: the creation of a body of design theory has been motivated mainly by the desire to exploit the power of the computer in making designs. Therefore, Simon argued for the establishment of a general “design science of the artificial” independent of specific application concerns. He challenged design theorists to “discover a science of design, a body of intellectually tough, analytic, partly formalizable, partly empirical, teachable doctrine about the design process” (pp. 131-132). Others (Alexander, 1964, 1979, 1996; Edmondson, 1987; Gross, Ervin, Anderson, & Fleisher, 1987; Schön, 1987; Newsome, Spillers, & Fingers, 1989) have taken similar positions on the study of design theory independent of specific fields.

Simon (1999) portrays the controlling logic of design as the formation and exploration of a set of alternative solutions that satisfy a set of constraints and criteria, and then selection of an alternative on the basis of a prioritizing rule. The efficient generation of multiple acceptable (“satisficing”, in Simon’s terminology) alternative solutions is a key activity of design that should be theory-driven if brute combinatorics and blind search are to be avoided. This is a clue to the nature of design theory: if the essential activity of technology is the creation of alternative structures, then the efficient generation of alternatives that in advance have some promise of being effective is a task that should require theoretical guidance. Design theories are, therefore, theories for use in structuring and synthesis (Gibbons, 2003a). Artifacts begin as conceptual entities, and the function of design theory is to supply the bridge between (a) conceptual entities and (b) workable artifact designs and plans for the construction of artifacts.

Design theories compete by being superior in achieving particular ends, measured in terms of one or more dimensions of outcomes. Multiple theories of multiple kinds are required, therefore, because criteria differ from problem to problem and theories are biased in terms of the range of artifacts they can produce, the outcomes they generate, and the side-effects that accompany them.

Domain Theory

Design theory can be contrasted with the domain theories of specific fields of design, such as engineering design, computer and computer chip design, architectural design, manufacturing design, structural design, and others. The most important result of improved domain theories may be the acceleration of advances in the quality and sophistication of designs, particularly in computer-aided design through modeling (Kuehlmann, 2003).

We categorize instructional theories as domain theories, similar in intent to the theories that have led to advances in these other fields. The theory domain of interest in instructional design is the acts that take place during an instructional conversation. Use of the word “theory” was at one time restricted to science, but technologists in general—including instructional technologists—have appropriated the word with increasing frequency and conviction to refer to design domain theory. Gage (1964), Bruner (1966), Snelbecker (1985), Gagné (1985), Oswald (1989), Reigeluth (1999), Merrill (Merrill & Twitchell, 1994), and others have made reference to instructional theory, as differing from learning theory, and have probed the nature and content of instructional theories. Still, many technologists hesitate to speak in terms of theory, being uncertain about what theory means when applied to a design technology rather than a science.

Simon (1999) engages in an extended discussion of the nature of technological (or design) theory and its differences from scientific theory. He explains, “The natural sciences are concerned with how things are…. Design, on the other hand, is concerned with how things ought to be, with devising artifacts to attain goals. We might question whether the forms of reasoning that are appropriate to natural science are suitable also for design” (pp. 114-5).

A Basis for the Design Theory/Domain Theory Distinction

The nature of scientific (or descriptive) theory is described as numerous bounded “local” theories and the hope that scientists might someday find a “theory of everything” (Hawking, 1998). Whether it is appropriate to consider a design “theory of everything” is a point for speculation, but we *can* speak in terms of local design theories and multiple varieties of local design theory. Vincenti (1990) provides insight into the kinds of theory that might be employed by a designer. He describes several categories of organized engineering design knowledge necessary for the solution of technological (or design) problems. They include: operational principles, normal configurations, criteria and specifications, intellectual concepts, mathematical tools, mathematically-structured knowledge, device-specific mathematical relationships, phenomenological theories, quantitative assumptions, quantitative data, practical considerations, and design instrumentalities.

All of these categories have importance to the discussion of design theory; each of them is a candidate to evolve in the future a body of synthetic (design-related) theory. Of these categories of design-related knowledge we will concentrate on two to outline their theoretical implications: operational principles, and design instrumentalities. We have selected these two because they deal with core concerns of designs—one with conceptual structures combined into designs, and the other with the processes by which they are brought together. These represent the sides of a gap that is bridged during design: the conceptual world in the designer's mind and the concrete world of designed artifacts. Next we will describe design instrumentality knowledge as it relates to instructional designs. In a later section we will return to the category of operational principle knowledge.

Design Instrumentalities and Instructional Design Theory

An enormous literature exists on design instrumentalities for instructional designers. However, the theoretic roots of current design practices are difficult to trace in that literature. The predominant formalism in the literature on instructional design is a collection of instructional planning methodologies that as a group are referred to as *ADDIE*, *ISD*, the *systems approach*, or *systematic development model*. These methods are purported to be derived from general systems theory, but the methods are often taught with a high degree of local variation without much reference to the foundational theory. This often includes an admixture of design processes with instructional theory, so that the design process appears to be theory-derived. The result has been a set of loosely-specified, non-standard, highly-variable design activities held up professionally more as an ideal than as a criterion, and that conflate the design process with specific domain theories of instruction.

On close examination, the practices of the systems approach appear to be a combination of practical project management considerations, instructional theory, and common sense. Andrews and Goodson (1991) document numerous examples of design and development models that are different combinations and orderings of a common set of design processes. It would not be exaggerating to say that hundreds or even thousands of these exist within training departments in industrial, commercial, government, military, and educational organizations as tailored local versions of a systematic process description.

Systematic instructional design is a process approach to design problem solving analogous to the waterfall process found early-on in other design fields but later de-emphasized. Such approaches are a way of breaking down large and complex design problems into more easily solved sub-problems. Simon (1999) and many others identify problem decomposition as an important step in problem solution and describe different ways in which a problem may be decomposed. Process decomposition is only one of these. The most prominent alternative to process decomposition is functional decomposition (Baldwin & Clark, 2000). Functional decomposition produces layered sub-problems that correspond to functions carried out by the designed artifact that enable it to fill its purpose. Brand (1994) describes this type of decomposition with respect to the design of buildings.

Design Layering

Brand (1994) describes the design of a building in terms of several integrated sub-designs, which he calls *layers*. Brand's layers of design represent solutions to design sub-problems created by decomposing the original design problem in terms of artifact functions. He therefore characterizes the complete design of a building in terms of multiple coordinated and integrated sub-designs. The layers, according to Brand are "fundamental to understanding how buildings actually behave" (p. 17). Each layer of a design performs one or more functions for the complete design. As the architect proceeds from drawing to drawing through layer after layer, Brand maintains, structures within layers must correspond across layers, and yet the layers are sufficiently independent of each other that changes to the design of one do not destroy the function of another. Baldwin and Clark (2000) refer to this as design modularization and provide an extended case study of how the functional design decomposition used in the design of the IBM 360 operating system revolutionized the design and economics of computers. Brand describes layers present in virtually all modern building designs, as shown in Figure 1.

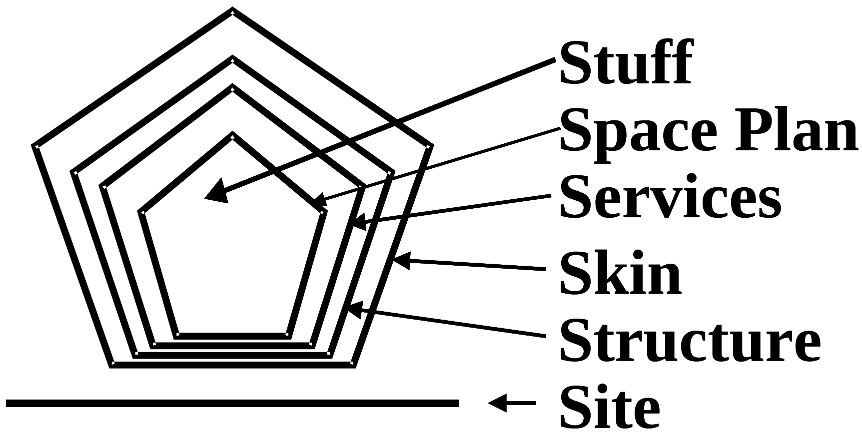

Figure 1. Brand's layers of building design. (From Brand, 1994.)

Brand defines the layers in this way:

- Site – The geographical setting and the legally defined lot, having boundaries and context
- Structure – The foundation and load-bearing elements of the building
- Skin – The exterior surfaces
- Services – The communications wiring, electrical wiring, plumbing, sprinkler system, HVAC (heating, ventilating, air conditioning), and moving parts like elevators and escalators
- Space plan – The interior layout—where walls, ceilings, floors, and doors go

- Stuff – Chairs, desks, phones, pictures, kitchen appliances, lamps, etc.: things that move around inside spaces (Brand, 1994, p. 13)

These layers have not always been considered part of building designs. The conception of design layers in a professional community may be interpreted as a measure of the maturity of a field of design.

Brand points out several important implications of the influence of layer awareness on designs:

- Layers age and change at different rates, but they can be designed and interfaced in a way that allows relatively independent, non-destructive change to individual layers.
- Layered design can therefore create artifacts that are adaptive and long-lived.
- The sequence of layers from "site" to "stuff" is the general sequence followed in both design and construction; moreover, it is related to the rate of aging of different layers. (Note: on this point we disagree with Brand, as we describe below.)
- Layers represent different sets of design skills with different agendas, design goals, and problems to solve and integrate.
- The dynamic of a building—the pace of change within and between layers—is dominated by the slowly-changing components; rapidly-changing components "follow along."

- Embedding layers together looks efficient but ultimately shortens the life of the building as changes become increasingly destructive.

Whether or not designers of buildings see their designs in terms of the layers that Brand describes is an important question. Certainly the trend of modern design standards supports designs that allow below-the-surface layers to be accessed through masking layers, repaired, and even changed with minimal disruption. Standard office building design clearly facilitates the reconfiguration of interior working spaces and the service layers behind them, and this design philosophy has spawned several systems of specialized tools, structural components, and construction methodologies. Examination of early housing designs in America shows that there was a period when simple construction took precedence over adaptability in designs. An innovation called “balloon construction” revolutionized housing design and produced consciously-layered designs early in the late 19th and early 20th centuries (Peterson, 2000). This standard set of layers is thrust upon designers in the form of received design practices in which a layering structure that has evolved over many years is implicit: a hint that the development of layers is a cross-generational phenomenon.

The evolution of layer awareness in housing designs seems to have gone through a series of predictable stages. Layering of designs occurs naturally as design criteria become more exacting and as design problems become more complex and demanding. The decisions and plans that could be made originally by an individual, multi-skilled person slowly fragment into local designs that involve the assistance of design specialists. Layers become evident in the design itself, which begins to consist of independent sub-designs that are integrated and orchestrated. New and more detailed sub-layers of the design come into existence through innovation. Eventually, as criteria continue to arise, a design team composed of specialists and coordinated by a lead designer is required in order to produce complete, consistent, and integrated designs.

Brand’s example of building layers is only one of many modern examples that can be provided of the maturation of a design field and the introduction of specialized, layered planning into designs. Additional examples can be found in the recent histories of computer chip design, software design, mechanical engineering of automobiles and aircraft, architecture, computer network design, and others (Baldwin & Clark, 2000; Kuehlmann, 2003; Saabagh, 1996). McCloud (1994) describes a principle of layering in relation to the design of comics. In many cases, rapid developments in a design field are made possible through the creation of design languages within layers that are amenable to computation, and the result is increasingly greater participation of the computer in design activities.

Design Layering and Instructional Design

Gibbons (2004) describes a set of layers derived from the functional properties of virtually all instructional designs. These layers represent specialized design sub-problems that result from the decomposition by functionality of whole instructional design problems. A representative set of instructional design problems is named and described by Gibbons (2003b):

- Content layer. A design must specify the structures of the abstract subject-matter to be taught, must identify the units into which the subject-matter will be divided, and must describe how elements of subject-matter will be made available to instructional functions performed by other layers.
- Strategy layer. A design must specify the physical organization of the learning space, social organizations of participants, their roles and responsibilities, instructional goals, allocation of goals to timed event structures, and strategic patterns of interaction between the learner and the instructional experience.
- Message layer. A design must specify the tactical language of message structures through which the instructional experience can communicate content-derived information to the learner in conversational form.
- Control layer. A design must specify the language of control structures through which the learner expresses messages and actions to the source of the learning experience.
- Representation layer. A design must specify the representations that make message elements visible, hearable, and otherwise sense-able: the media

representation channels to be used, the rule for assigning message elements to media channels, the form and composition of the representation, the synchronization of messages delivered through the multiple channels, and the representations of content.

- Media-logic layer. A design must specify the mechanism by which representations are caused to occur in their designed or computed sequence.
- Data management layer. A design must specify data to be captured, archived, analyzed, interpreted, and reported.

The concept of design layers constitutes a structuring theory for the creation of instructional designs. Each layer accounts for a certain category of decisions regarding specialized functions that eventually become part of a complete design. The division of layers we present is not scientifically derived, and it is not presented as a “truth”. Layers, especially at the more detailed levels of design, evolve and change based on their utility to the designer according to a number of factors that include design constraints, criteria, resources, tools, technology, construction methods, and available designer skills. The list of layers we suggest is generic to virtually all instructional design projects, but one arrangement of specific sub-layers may be superior to another and confer advantage on a designer. What is emphasized here is not the power of the particular set of layers we have enumerated, but the power of thinking of instructional designs in terms of layering. We believe that it represents a way to advance thinking about the properties of instructional designs and the relationship between instructional theory and instructional design theory.

Design Languages

Schön (1987) refers to layers as domains of language: “Elements of the language of designing can be grouped into clusters, of which I have identified twelve…. These design domains contain the names of elements, features, relations, and actions and of norms used to evaluate problems, consequences, and implications.” (p. 58). He continues: “Aspiring members of the linguistic community of design learn to detect multiple references, distinguish particular meanings in context, and use multiple references as an aid to vision across design domains” (p. 61). Gibbons and Brewer (2005) and Waters and Gibbons (2004) describe in detail design languages and the notation systems that make them public and shareable.

Natural Languages and Design Languages

Natural languages are typified by a set of primitives, a syntax, and a semantic (Berlinski, 2000; Cooke, 2003; Jackendoff, 2002). Table 2 highlights differences between natural languages and design languages in these respects. The terms of a natural language tend to evolve from usage, as objects and events are encountered repeatedly in everyday experience, sufficiently to where an abstraction of them is formed and given a name or symbol. General social use of the terms over time brings them into the language. Design languages exist as tools for problem solving and design synthesis. Their expressions have meaning only within the domain of problems for which they were created.

Table 2. Natural languages and design languages compared in terms of primitives, syntax, and semantics.

	Natural Language	**Design Language**
Primitive terms	Centered in everyday things and events; abstractions of experience	Centered in tools, processes, technologies, theories, or best practices of a domain
Syntax	Based on words as a medium of expression in which linear or positional order is critical	Dependent on the medium of problem solving and solution; sometimes time, space, or view-oriented
Semantics	Derived from the world as it is experienced and things that can be, or are desired to be, communicated	Derived from the problem domain, the context of problems in the domain and available technologies

Abstraction of and Naming of Design Concepts

The vast majority of designs employ structures "borrowed" or abstracted from previous designs that can be characterized as the terms of a design language. Alexander (1979) describes the abstraction of architectural patterns—a pattern language—from buildings for the purpose of applying those patterns in later designs. "A pattern describes a problem that occurs over and over again in our environment, and then describes the core solution to that problem, in such a way that you can use this solution a million times over" (Alexander, 1979, p. x).

Vincenti (1990) names among his classes of specialized technological knowledge, classes like *Operational Principles*, *Normal Configurations*, and *Intellectual Concepts* that are closely related to design languages: ”Conceiving and analyzing artifacts requires thoughts in people's minds…. Intellectual concepts [and operational principles and normal configurations] provide the language for much of such thinking (p. 215).

Design Languages and the Advance of Design Practice

Rapid advances in the sophistication, effectiveness, productivity, and quality of designs have been made possible by the cultivation of improved design languages. Most often this accompanies the automation of design processes. For this reason, advances over the past three decades in computer-assisted design (CAD) and computer-assisted manufacturing (CAM) can be attributed to the discovery of specialized languages for problem expression and the representation of solutions whose terms can be translated into languages that are computable (Kalay, 1987; Newsome et al., 1989).

Early CAD/CAM systems did little more than capture data entered into them by a human designer: the software had no ability to recognize higher-order abstractions and no ability to make computations in terms of groupings of lines that might represent a building wall or a hydraulic coupling. As abstractions for such groupings were introduced into the design languages of these programs, the programs could begin to reason about them, making more and more decisions about them as an abstract unit of the design.

The literature documenting the evolution of automated computer chip design systems shows that local problems came under automation or semi-automation as local languages were invented that conveyed to the computer the elements of the problem and the elements of solutions for design sub-problems. Today, the great majority of routine design decisions during chip design are made by the computer, and as a result, much more complex and powerful designs have become possible, while design time has been cut significantly.

We have dealt in this section with how Vincenti's *design instrumentalities* category of technological knowledge anticipates a body of theory related to making instructional designs—instructional design theory.* We have proposed a layer theory of design structure that is based on an alternative approach to design problem fragmentation that uses artifact function rather than process as the decomposition principle. We have further proposed that layers are defined in terms of multiple design languages used for the solution of layer and sub-layer design problems. In the section that follows we will propose that another of Vincenti's categories, *operational principles*, anticipates a different type of theory that describes how designs work—instructional theory.

Operational Principles and Instructional Theory

* Editors' note: This is a combination of what we call instructional-planning theory and instructional-building theory (see Chapter 1). [Author's note: I have come to see much value of this distinction, but at this late date prior to publication I would like to retain my current terminology.]

Instructional theories are a major source of design languages (other sources being traditional practice, standard-setting, metaphorical extension, popular discourse, and insight and invention).

Design Languages and Operational Principles

Vincenti's category of technological knowledge called *operational principles* is of special importance to linking design layers and design languages to instructional theory. It supplies abstractions that create a semantic context for design language terms and therefore for central structural elements of instructional theory.

An operational principle, according to Polanyi (1958) is part of the "logic of contriving". This logic describes how a human-made artifact works:

> *There is a specifiable* reason *for every step of the procedure and every part of the machine, as well as for the way the several steps and the various parts are linked together to serve their joint purpose. This chain of reasons is set out in the operational principles of the process or of the machine (p. 332).*

Operational principles are abstract descriptions of the oppositions and coordinations of dynamic forces that can be incorporated into human-designed artifacts—the essential inner workings of functioning artifacts. They describe those workings—the transmission and transformation of energy and information—independent of specific material form. Operational principles have generative power for the design of artifacts: specific dimensions and materials are assigned during design to the abstract elements of one or more operational principles.

An operational principle is implemented through substitution:

> *Just as the rules of algebra will operate for any set of numbers for which the algebraic constants may stand, so an operational principle applies to any collection of parts which are functioning jointly according to this principle. (Polanyi, 1958, p. 329)*

A single operational principle can be used to generate multiple artifact configurations through the substitution of specific mechanisms and materials in place of the abstract elements that make up the principle. Layton (1992) explains that designs are made by assigning specific materials and dimensions to conceptual structures that represent abstract relationships of elements. Layton notes that the design activity of *assigning* dimensions to an abstraction differs from the activity of science, which attempts to discover relationships *as free as possible* of specific dimensions.

Layers, Languages, Operational Principles, and Instructional Theory

We propose that what an instructional theorist expresses in an *instructional theory* is a set of specialized, mutually-consistent design languages, consisting of terms the theorist defines, that are distributed across multiple design layers which are defined by an *instructional design theory*. Instructional design theory provides a structural framework of layers within which specific instructional theories can be analyzed and compared. To the extent that different observers can agree upon a common definition of layers, they can jointly and publicly carry out such analyses and comparisons.

The outward form of an instructional theory consists of verbal propositions that relate the design language terms the theorist has chosen to define. Through these propositions we can see a set of operational principles held by the theorist that express the major assumptions—the real fabric—of the instructional theory. The operational principles underlying an instructional theory, and the categories and propositions of the theorist, provide a generative mechanism capable of creating multiple instructional artifact designs which on the surface differ in form but under the surface share a common architecture. Several new-paradigm instructional examples were reviewed by Gibbons and Fairweather (2000) and shown to possess a similar underlying architecture, described by a single operational principle they called “model-centered instruction” (see also Gibbons, 2001).

Table 3 presents a layer-by-layer comparison of three well-known instructional theories: John R. Anderson’s theory of intelligent tutoring (Anderson, Corbett, Koedinger, & Pelletier, 1995), cognitive apprenticeship (Collins, Brown, & Newman, 1989), and Gagné’s theory of the conditions of learning (Gagné, 1985). These theories were chosen because they are clearly expressed, are widely known, and have a history of extensive application. Table 3 shows that each theory defines a set of design language terms within one or more design layers.

Table 3. Analysis of some well-known instructional theories to show the relationship of instructional theories to the framework provided by layers, which have their basis in instructional design theory.

Theory	Anderson	Cog App	Gagné
Content layer	Content subdivided into two types: “production rules” and semantic units called “working memory elements”	Four content types: -Domain knowledge -Problem solving strategies and heuristics -Control strategies -Learning strategies	Taxonomy divides knowledge into 5 main types; one type, intellectual skills, is subdivided into several sub-categories
Strategy layer	-Production rules learned in prerequisite order -Learning by practice and error correction	6 methods: -Modeling -Coaching -Scaffolding -Reflection -Articulation -Exploration 5 social strategies: -Situated learning -Culture of expert practice -Intrinsic motivation -Exploit competition -Exploit cooperation 3 sequencing strategies: -Increasing complexity -Increasing diversity -Global before local	Conditions to support learning are determined by the type of knowledge to be learned; nine events of instruction provide occasions for those conditions to be expressed
Control layer	Control resides in the system; student responds to problems presented	Implied in apprentice interpersonal relationships, but not enumerated	Implied instructor control; student responds to instruction
Message layer	No formalization of message structuring guidelines	No formalization of message structuring guidelines	Types of message used in illustrations, but no formalization of messaging guidelines

Representation layer	No formalization of representation terms or guidelines	No formalization of representation terms or guidelines	Types of representation used in illustrations, but no formalization of representation terms or guidelines
Media-Logic layer	No formalization of media-logic guidelines	No formalization of media-logic guidelines	No formalization of media-logic guidelines
Data management layer	Data management specified as use of data from previous responses to influence future selections of the system regarding problems to present	No formalization of data management guidelines	No formalization of data management guidelines

Anderson's instructional theory contains propositions concerning the organization of the content layer of designs. The theory is based on the assumption of two types of knowledge: production rules and working memory elements. Cognitive apprenticeship defines four categories of knowledge, implying that the result of an analysis of content structures will be expressed in terms of these categories. Though the categories are identified, specific propositions that link categories to strategic patterns are not given. In contrast, Gagné's division of learnable content into five major categories and one of those categories (intellectual skills) into several sub-categories is closely linked with the central premise of Gagné's theory—that specific content types can be used to bound instructional strategy design.

Most importantly, all three instructional theories take a position on the nature of content and the appropriate categories into which it is partitioned. A designer who agrees with a theorist's partitioning of content can use the theory—

and the content design language of terms it supplies—for analysis purposes. Gibbons and his associates (Gibbons, Nelson, & Richards, 2000a, 2000b) provide a review of the basic principles of pre-design analysis that considers in some detail the design issues of the content layer.

The three theorists compared in Table 3 differ also with respect to the structures and languages that they propose at the strategy layer of designs. Anderson’s theory, as already noted, closely links content structures with interactions, and curriculum tends to be centered on a body of rules practiced in a calculated sequence. Cognitive apprenticeship does not link specific content types with specific instructional methods. However, the theory specifies a great deal more structure at the strategy layer than either Anderson or Gagné. In addition to describing six instructional methods, cognitive apprenticeship describes alternative social organizations (expert practice culture) and employment of social forces (exploit competition, exploit cooperation) for instructional purposes where the other theories are largely silent. The 18 principles of cognitive apprenticeship under four layer-like headings as summarized in Table 1 of Collins, Brown and Newman (1989) are mostly expressed in a form that reveals the abstract operational principle from which a large family of very different designed surface forms can be generated. In addition to organizations of social forces, cognitive apprenticeship design language terms support the design of instructional sequences.

Gagné’s theory links methods of instruction with learning types, as already mentioned. In addition, Gagné describes nine events of instruction that further define those methods. His theory does not give focus to the social dimensions of organization, but a broad outline of assumptions about instructor and learner roles is evident, which is described by Gibbons et al (in press). The nine events are not described as sequencing constructs, and a caution is given that the structures of the nine events are not meant to correspond with distinct slices of time. However, many of the events described by Gagné have a temporal relationship that is hard to avoid.

The three theorists compared in Table 3 say little about the structuring of designs at the remaining layers. We do not feel that this is due to the unimportance of these layers to the theorists but to the immediate purpose of the author in writing and the critical issues the author is trying to bring into focus; the most attention is given to layers the theorists consider most important. We take this as implicit evidence that design involves the use of multiple local theories related to layer-specific concerns rather than single monolithic theories, as is sometimes implied in the instructional design literature.

Table 4 shows that other theorists have given attention to different layers.

Table 4. Sampling of work by theorists or research reviewers attempting to identify layer-specific principles.

Layer	Theorist/Author	Principles
Control	Crawford (2003)	Conversational interaction and the design of interfaces to support rich user communication and conversation with the system
	Gibbons & Fairweather (1998)	Varieties of human-machine communication (learner to system) during instruction and the computer's ability to implement them
Message	Merrill (1994)	Categorization of message elements that make up an instructional strategy; texturing principles that prioritize certain messages and foreground certain information
	Horn (1997)	Categorization and logical grouping of information tableaus; emphasis on underlying relationships within message groupings rather than on their display
	Simon & Boyer (1967)	Compendium of analysis methods for describing student-teacher communications and interpretable actions during classroom instruction
Representation	Mayer (2001)	Principles for the use of synchronized multimedia channels to convey instructional information in a manner that supports learner formation of appropriate mental models
	Tufte (1990, 1997)	Principles for the use of graphical representations to present complex and dynamic bodies of information
	Wurman (1997)	Visual designers explain and illustrate their principles for explaining using visual and textual structure
	Harris (1999)	Varieties of presentation of data in graphical form and principles for constructing data representations
	Fleming (1993)	Message design principles, concentrating on the representation of information
Media-Logic	Gibbons et al. (2001)	Principles of merging media structures with other design structures
	Seels et al (1996)	Principles related to the design of instruction involving the television medium; extensive glossary of terms, many of which are the terms of a specialized design language
	Hannafin et al (1996)	Principles related to the design of computer-based instruction as a medium
	Romiszowski & Mason (1996)	Principles related to the design of computer-mediated communication
	Stanney (2002)	Principles related to the design of virtual environments
Data Management	Wenger (1987)	Summary of intelligent computer-based instruction design principles, including use of data to create adaptive instruction

	Stolurow (1969)	Early conception of the principles for the use of data from instructional interactions to determine the future path of instructional events; dated by reference to programmed instruction but relevant in principle

These authors and works are merely suggestive of the layer-relatedness of an enormous body of writing on design principles. Some layers, such as Control, are underrepresented in the design theory literature. However, the control layer has become more central to designers as interest has increased in video games, instructional simulations, and microworlds, in which control systems are necessary for user navigation of complex information, physical, and problem-solving spaces.

The Message layer is also under-represented in current literature, despite the recent emphasis on social interaction during instruction. However, Sawyer (2006) reviews systems for message structuring and describes the early interest in this area of design language (See also Simon & Boyer, 1967). Messaging in most media is accomplished using pre-composed display content (combined graphics and text, animations, or video). However, instructional messages in the future will increasingly be composed at the moment of use from a variety of sources. This trend already supplies the competitive edge for non-instructional marketing Web sites. Seen in this perspective, the deliberate design of messaging patterns that can be filled with specific representation content at the moment of need from diverse sources plays an important intermediary role in assembling the raw elements during the construction of displays. Viewed in this light, Merrill's component display theory (Merrill, 1994), often viewed as a formula for designing instructional strategies, can be seen as a type of message design language for constructing individual messages in the service of a learner- or system-initiated instructional strategy. Message design languages identify message tokens that can be used to carry the intentions of instructional communications, without describing the exact content of the representations.

Representation theories and their associated languages are invisible to most designers because representation technology has for so long been confounded with message and media-logic concerns for pre-composing and storing display content. However, recent innovations in representation technologies provide the designer with more options for the display of information, sometimes using data supplied at the moment of need to generate specific display content. As greater amounts of the display are created or arranged at the moment of use, the principled design of representations will take priority over the storing of individual representations.

Media-logic design languages are introduced with each new medium, tool, or technique. Media production is the nexus of the most commonly-known instructional design languages, and numerous detailed glossaries and lexicons of such languages are abundant in libraries and on the Web.

Data management layer concerns have become muted as the goals of adaptive and generative instruction have been subordinated over the past three decades to productivity and lowered costs. Stolurow (1969) describes the ideal of adaptive instruction in terms of programmed instruction-like products. Though the fashion in structures manipulated during instruction today has changed, the principle of adaptivity in Stolurow's writing remains unchallenged. Wenger (1987) describes early experiments in adaptive instruction and provides numerous examples of ways in which data resulting from instructional interactions were used to select and sequence future instructional events. As interest in adaptive instruction, adaptive curricula, and adaptive instructional organization increases, the design languages for designing data management systems will become more important.

Conclusion

Our purpose has been to describe a particular view of the architecture of instructional theory, framed within an instructional design theory of function-related design layers. We have related the separation of these bodies of theory to a similar separation that has occurred in other design fields. This more detailed framework for theoretical ideas describes design decision-making at a finer granularity and concentrates on the functional characteristics of the designed artifact, rather than on the design process.

We propose that this layered architecture of instructional theory will accomplish the following: it will give designers a tool to create quality designs more consistently, it will facilitate communications about designs and theories, it will allow designers to work efficiently in design teams with a greater degree of mutual understanding, it will suggest functionalities for more advanced and productive design tools, and it will allow experienced designers to communicate design knowledge and judgment to novices more quickly.

References

Alexander, C. (1964). *Notes on the synthesis of form*. Cambridge, MA: Harvard University Press.

Alexander, C. (1979). *The timeless way of building*. New York: Oxford University Press.

Alexander, C. (1996). The origins of pattern theory, the future of the theory, and the generation of a living world. *IEEE Software, 16*(5), 71-82.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: lessons learned. *Journal of the Learning Sciences, 14*(2), 167-207.

Andrews, D. H., & Goodson, L. A. (1991). A Comparative analysis of models of instructional design. In G. J. Anglin (Ed.), *Instructional Technology; Past, Present, and Future* . Engelwood, CO: Libraries Unlimited.

Baldwin, C. Y., & Clark, K. B. (2000). *Design Rules, Vol. 1: The Power of Modularity*. Cambridge, MA: MIT Press.

Berlinski, D. (2000). *The Advent of the Algorithm*. New York: Harcourt.

Bernstein, L. (1976). *The Unanswered Question: Six Talks at Harvard (The Charles Eliot Norton Lectures, 1973)*. Cambridge, MA: Harvard University Press.

Bichelmeyer, B. A., Boling, E., & Gibbons, A. S. (2006). Instructional Design and Technology Models: Their Impact on Research and Teaching in IDT.

In M. Orey, J. McClendon, & R. M. Branch (Eds.), *Educational Media and Technology and Media Yearbook, 2006* (Vol. 31,). Westport, CT: Libraries Unlimited.

Brand, S. (1994). *How buildings learn: What happens after they're built*. New York: Penguin Books.

Bruner, J. (1966). *Toward a Theory of Instruction*. Cambridge, MA: Belknap Press.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* . Hillsdale, NJ: Lawrence Erlbaum Associates.

Cooke, D. (2003). *A Concise Introduction to Computer Languages: Design, Experimentation, and Paradigms*. Pacific Grove, CA: Brooks/Cole Publishing.

Crawford, C. (2003). *The Art of Interactive Design*. San Francisco, CA: No Starch Press.

Edmondson, A. C. (1987). *A Fuller Explanation: The Synergetic Geometry of Buckminster Fuller*. Boston, MA: Barkhauser.

Fleming, M. L. (1993). *Instructional message design: Principles from the behavioral and cognitive sciences (2nd ed.)*. Englewood Cliffs, NJ: Educational Technology Publications.

Gage, N. L. (1964). Theories of teaching. In E. R. Hilgard (Ed.), *Theories of Learning and Instruction: The Sixty-third Yearbook of the National Society for the Study of Education* (pp. 269-285). Chicago: University of Chocago Press.

Gagné, R. M. (1985). *The Conditions of Learning (4th ed.)*. New York: Holt, Rinehart & Winston.

Gibbons, A. S. (1998). Model-centered instruction. *Journal of structural learning and intelligent systems,* 14(4), 511-540.

Gibbons, A. S. (2003a). The practice of instructional technology: Science and Technology. *Educational Technology, 43*(5), 11-16.

Gibbons, A. S. (2003b). What and how do designers design?: A theory of design structure. *Tech Trends, 47*(5), 22-27.

Gibbons, A. S. (2004, June 3-4, 2004). *The interplay of learning objects and design architectures.* Paper presented at the Partnership in Global Learning Workshop on e-Learning Objects and Systems, Orlando, FL.

Gibbons, A. S., & Brewer, E. K. (2005). Elementary principles of design languages and notation systems for instructional design. In J. M. Spector, C. Ohrazda, A. Van Schaack, & D. Wiley (Eds.), *Innovations in Instructional Technology: Essays in Honor of M. David Merrill* . Mahwah, NJ: Lawrence Erlbaum Associates.

Gibbons, A. S., & Fairweather, P. G. (1998). *Computer-Based Instruction: Design* *and Development*. Englewood Cliffs, NJ: Educational Technology Publications.

Gibbons, A. S., Lawless, K. A., Anderson, T. A., & Duffin, J. R. (2001). The web and model-centered instruction. In B. R. Khan (Ed.), *Web-based training* . Englewood Cliffs, NJ: Educational Technology Publications.

Gibbons, A. S., Merrill, P. F., Swan, R., Campbell, J. O., Christensen, E., Insalaco, M., & Wilcken, W. (in press). Re-examining the implied role of the designer. *Quarterly Review of Distance Education.*

Gibbons, A. S., Nelson, J., & Richards, R. (2000a). *Theoretical and practical requirements for a system of pre-design analysis: State of the art of pre-design analysis* (White Paper). Idaho Falls, ID: Center for Human-Systems Simulation, Idaho National Engineering and Environmental Laboratory (DOE).

Gibbons, A. S., Nelson, J., & Richards, R. (2000b). Model-centered analysis process (MCAP): A pre-design analysis methodology (White Paper). Idaho Falls, ID: Center for Human-Systems Simulation, Idaho National Engineering and Environmental Laboratory (DOE).

Gross, M., Ervin, S., Anderson, J., & Fleisher, A. (1987). Designing with constraints. In Y. E. Kalay (Ed.), *Computability of design* . New York: John Wiley & Sons.

Hannafin, M. J., Hannafin, K. M., Hooper, S. R., Rieber, L. P., & Kini, A. S.

(1996). Research on and research with emerging technologies. In D. H. Jonassen (Ed.), *Handbook of Research for Educational Communications and Technology* . New York: Macmillan Library Reference USA.

Harris, R. L. (1999). *Information Graphics: A Comprehensive Illustrated Reference*. Oxford, UK: Oxford University Press.

Hawking, S. (1998). *A Brief History of Time*. New York: Bantam.

Horn, R. E. (1997). Structured writing as a paradigm. In C. R. Dills, Romiszowski, A. J. (Ed.), *Instructional Development Paradigms* . Englewood Cliffs, NJ: Educational Technology Publications.

Jackendoff, R. (2002). *Foundations of Language: Brain, Meaning, Grammar, Evolution*. Oxford, UK: Oxford University Press.

Kalay, Y. E. (1987). *Computability of Design*. New York: John Wiley & Sons.

Kuehlmann, A. (Ed.). (2003). *The Best of ICCAD : 20 Years of Excellence in Computer-Aided Design*. New York: Springer.

Layton, E. (1992). Escape from the jail of shape: Dimensionality and engineering science. In P. Kroes & M. Bakker (Eds.), *Technological development in the industrial age* . Dordrecht, The Netherlands: Kluwer Academic Publishing.

Mayer, R. E. (2001). *Multimedia Learning*. Cambridge, UK: Cambridge University Press.

McCloud, S. (1994). *Understanding Comics*. New York: Harper Perennial.

Merrill, M. D. (1994). The descriptive component display theory. In M. D. Merrill & D. G. Twitchell (Eds.), *Instructional Design Theory* . Englewood Cliffs, NJ: Educational Technology Publications.

Merrill, M. D., & Twitchell, D. G. (Eds.). (1994). *Instructional Design Theory*. Englewood Cliffs, NJ: Educational Technology Publications.

Newsome, S. L., Spillers, W. R., & Finger, S. (1989). *Design Theory '88: Proceedings of the NSF Grantee Workshop on Design Theory and Methodology*. New York: Springer-Verlag.

Oswald, D. F. (2002). A conversation with Glenn E. Snelbecker. *Educational Technology, 42*(5), 59-62.

Peterson, F. W. (2000). Anglo-American wooden frame farm houses in the midwest, 1830-1900: Origins of balloon frame construction. In S. A. McMurry (Ed.), *People, Power, Places* . Knoxville: University of Tennessee Press.

Polanyi, M. (1958). *Personal Knowledge: Towards a Post-Critical Philosophy*. New York: Harper Torchbooks.

Reigeluth, C. M. (1999). *Instructional-design theories and models, Volume II: A new paradigm of instructional theory* (Vol. II). Mahwah, NJ: Lawrence Erlbaum Associates.

Romiszowski, A. J., & Mason, R. (1996). Computer-mediated communication. In D. H. Jonassen (Ed.), *Handbook of Research for Educational*

Communications Technology . New York: Macmillan Laibrary Reference USA.

Saabagh, K. (1996). *21st-Century Jet: The Making and Marketing of the Boeing 777*. New York: Scribner.

Sawyer, R. K. (2006). Analyzing collaborative discourse. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences.* Cambridge, UK: Cambridge University Press.

Schön, D. A. (1987). *Educating the Reflective Practitioner*. San Francisco, CA: Jossey-Bass Publishers.

Seels, B., Berry, L. H., & Horn, L. J. (1996). Research on Learning from Television. In D. H. Jonassen (Ed.), *Handbook of Research for Educational Communications Technology* . New York: Macmillan Library Reference USA.

Simon, A., & Boyer, E. G. (Eds.). (1967). *Mirrors for Behavior III: An Anthology of Observation Instruments*. Wyncote, PA: Communication Materials Center.

Simon, H. A. (1999). *The sciences of the artificial (3rd ed.)*. Cambridge, MA: MIT Press.

Snelbecker, G. E. (1985). *Learning Theory, Instructional Theory, and Psychoeducational Design*. Lanham, MD: University Press of America.

Stanney, K. M. (Ed.). (2002). *Handbook of Virtual Environments: Design,* *Implementation, and Applications*. Mahwah, NJ: Lawrence Erlbaum Associates.

Stolurow, L. M. (1969). Some factors in the design of systems for computer-assisted instruction. In R. C. Atkinson & H. A. Wilson (Eds.), *Computer-Assisted Instruction: A Book of Readings* . New York: Academic Press.

Tufte, E. R. (1990). *Envisioning Information*. Cheshire, CT: Graphics Press.

Tufte, E. R. (1997). *Visual Explanations: Images, Quantities, Evidence, and Narrative*. Cheshire, CT: Graphics Press.

Vincenti, W. G. (1990). *What engineers know and how they know it: Analytical studies from aeronautical history*. Baltimore, MD: Johns Hopkins University Press.

Waters, S., & Gibbons, A. S. (2004). Design languages, notation systems, and instructional technology: A case study. *Educational Technology Research and Development, 52*(2), 57-69.

Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge*. Los Altos, CA: Morgan Kaufmann Publishers.

Wurman, R. S. (1997). *Information Architects*. New York: Graphis.