



All Faculty Publications

2016

The Application of Layer Theory to Design: The Control Layer

Andrew S. Gibbons III

Brigham Young University, andy_gibbons@byu.edu

Matt Langton

BYU

Follow this and additional works at: <https://scholarsarchive.byu.edu/facpub>

 Part of the [Educational Methods Commons](#), [Educational Psychology Commons](#), and the [Educational Technology Commons](#)

Original Publication Citation

Gibbons, A. S. & Langton, M. B. (2016). Application of layer theory to design: The control layer. *Journal of computing in higher education*, 28(2), 97-135.

BYU ScholarsArchive Citation

Gibbons, Andrew S. III and Langton, Matt, "The Application of Layer Theory to Design: The Control Layer" (2016). *All Faculty Publications*. 2968.

<https://scholarsarchive.byu.edu/facpub/2968>

This Peer-Reviewed Article is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in All Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.

The Application of Layer Theory to Design: The Control Layer

Andrew S. Gibbons
Matthew B. Langton
Brigham Young University

Abstract

Validation of an architectural theory of instructional design layering is accomplished for one of the proposed layers by verifying the theory's claim that for every layer there exists a body of design theory from outside the field of instructional design that is capable of informing design within that layer.

*"Often the main value of a theory lies in the new kinds of research it generates."
N. L. Gage (1964, p. 281)*

Prologue

In 1985 Stephen Hawking lost the last vestiges of his voice, and his communication channel to the outside world nearly closed. All of Hawking's motor functions had deteriorated due to motor neuron disease to the point where even the simplest movements required strenuous effort, and in 1985 he had a tracheotomy that took away his powers of speech. This seemed to be the end of a brilliant career in cosmological physics. Incredibly, in the years since losing his voice, and in the face of advancing incapacitation, Hawking's scholarly output has actually increased:

On his own website, Hawking lists 194 publications through November 2008 (<http://www.hawking.org.uk>). This more complete list includes items such as books, significant lectures, and award essays. (Lehman, 2011, p. 6)

Since 2011, the year of the author's 69th birthday, there are additional papers, an autobiography, and two novels for young readers on cosmology—the latest in a series of novels written with his daughter. (Lehman, 2011 gives many more details of Hawking's writing history.)

This remarkable record of productivity, which amounts to millions of words of output, had to be achieved through the use of a control system operated by a single command action: the pressing of a switch. This system is described by Mehta in a chapter appropriately titled, "When a Single Button is All That Connects You to the World" (Mehta, 2007; see also Medeiros, 2015).

The control switch itself that operates Hawking's editing system is not the innovation. Later, when pressing a switch became impossible, the twitch of a facial muscle was used as a control. The key to Hawking's expressive control consists of an anticipatory, feed-forward interpretation system capable of using the control operation and its context to predict expressive intent, and by doing so offer appropriate shorthand control options that multiply the possible meanings of a single control actuation.

The design of this control system consists of designs for everything from the switch/twitch responder to the text editor interface and all of the predictive software in between. The design of this system is an example of the application of theoretical principles for control design. Over the years, specific hardware and software parts of Hawking's system have come and gone, but the theoretical principles underlying the design have remained the same. Hawking's case is one of many cases that demonstrate that the design of control systems is, or can be, more challenging and more theoretical than one would think.

Purpose

This paper reports research on the validity of a theory of design layers. Gibbons (2014) proposes that individual functional layers of a design constitute architectural features of an instructional design through which theoretical principles can influence designs. Specifically, this study examines a single functional layer--the control layer--in terms of theories of control design borrowed from disparate design fields. It argues that principles for instructional control design can be derived from the theories taken from these other fields and applied beneficially into instructional designs.

The results of this study appear to support the position that layers do constitute a valid channel for theoretical input to design for the following reasons:

- Theories for control design from other fields do contribute principles that can be abstracted and applied to the design of instructional control systems.
- These principles focus designer attention more directly on issues of control system design by pointing out considerations that might otherwise have been overlooked by instructional designers.
- Applying these principles improves the operation and the desirable qualities of the designed artifact.

As designs in other fields have become more sophisticated, control design theories have been shown to be important for designing control systems with desirable properties discussed in this paper. Maturing intellectual concepts of instructional design and rising consumer expectations of instructional experiences will in the future make it necessary for instructional designers to become more familiar with the theoretical bases of their control system designs, a pattern that has already taken hold in other design fields.

Method

Our method has been:

1. To examine the range of controls used in instructional designs to bring into focus the extent of the control design problem.
2. To identify a list of desirable properties of control systems found in other design fields.
3. To identify theoretical principles for control design from other design fields.
4. To extrapolate the application of those theoretical principles to instructional control system design, showing how theory can lead to new insights for instructional designers.
5. To identify instances where attention to control design theory has improved the quality and effectiveness of designed artifacts.

The control layer was originally selected for this study with expectation that it would provide a relatively uncomplicated exploration of control design theories and their proper application through design layers. Emphasis was on establishing the control layer as a legitimate area of design focus, therefore adding validity to the theory of design layers itself. As the study progressed, it became apparent that: (1) control systems for instructional purposes were much more complex and varied than we had expected, especially for more advanced forms of instruction, (2) the desirable properties of control systems in general, and by extrapolation of instructional control systems, were more numerous and exacting than we had anticipated, (3) the number of contributing theories from other design fields were more numerous and more fully evolved than we had expected, (4) that the large number of applicable theoretical principles makes it infeasible to describe more than a handful of examples, rather than the more exhaustive treatment we had hoped for, and (5) that the number of examples of control design theory contributions to design effectiveness is also so large as to prevent a full description.

In brief, we found that the layer we had chosen--the one we thought would lead to a quick and simple case study--was one with practical implications that reached into the very core of instructional conversation design--the design of interactive and adaptive instructional experiences. The volume of control design theories and principles in virtually every other design field lengthened this study by an additional year, and we still feel like novices on this large subject.

Assumptions

Our primary assumption is that applying theory improves designs. If this assumption is not correct, then teaching theory to novice designers has no purpose. If, on the other hand, instructional designs are improved by applying theory, then the manner by which theory is applied to designs should be studied in as much depth as possible.

A second assumption is that what we normally refer to as instructional “designs” are not monolithic entities, but rather integrated collections of sub-designs, interfaced to operate together harmoniously (Brand, 1994; Gibbons, 2014). The functional decomposition of design problems described by Gibbons is common in almost all design fields. It is described in the literature of architecture (Brand, 1994), computer design (Baldwin & Clark, 2000), business (Martin, 2009), and software design (Fowler, 1999). What is new is the application of this principle to instructional experience design. The control design layer which is the subject of this paper is just one of the functional layers proposed by Gibbons in an architectural theory of instructional design (Gibbons, 2014). It represents a sub-design problem that can be solved semi-independently and integrated with other solutions from other layers. The principle of modular design applies to instructional designs as well as to designs in other fields.

Theory About Theory

The use of the term “theory” in this study requires clarification. The term is used in two senses here: (1) to refer to *scientific* theory as traditionally defined (see, for example, Dickmeyer, 1989), and (2) to refer to *technological* theory as it has been more recently defined (Simon, 1999; Venable, 2006; Carlisle, 1997; Vincenti, 1990; Constant 1984). Scientific theory is used by scientists to summarize and *explain* observed phenomena. Technological theory is used by designers to *imagine* and *devise* tools and processes that do not yet exist. Dorst (2010, 2015) suggests that this relies heavily on abductive reasoning (see also Klir, 1969; Walton, 2005; and Josephson & Josephson, 1996).

Scientific theory is affirmed by repeated experimental confirmation of the predictions of models that scientists build. Scientific theory is capable of falsification. Two competing theories attempting to explain the same phenomenon are pitted against each other until research demonstrates that one provides the better explanation. (Consider, for example, the struggle between the Copernican and Ptolemaic models of the solar system.)

Technological theory is not capable of falsification. Many different theories can contribute to the evolution of a single design. Technological theory suggests structures to a designer that are capable of harnessing and diverting natural energies and information for human purposes. Scientific theory may describe a complex relationship of entities and variables without suggesting how that relationship can be used. Therefore, technological theory guides decision making and action in a way that scientific theory cannot accomplish. Scientific theory does not translate directly into technological theory, and much technological theory evolves well before scientific theory is invented to explain how and why things work.

Educational theorists like Bruner (1966), Gage (1964), and Dewey (1929) argue in favor of the distinction between scientific and technological forms of theory. Dewey observed that “The sciences of mechanics and mathematics are, in themselves, the sciences which they are, not sciences of bridge building. They *become* the latter when selected portions of them are focused upon the problems presented in the art of bridge building” (p. 35). From the perspective of general systems theory Klir (1969) argues the importance of this difference, showing that though both science and engineering study systems, they are starting from different perspectives and create different types of knowledge (see also Vincenti, 1990). The symbiotic relationship of science and technology in human knowledge creation is described by Gibbons and Bunderson (2005). The translation from scientific to technological theory is non-trivial and includes a process normally referred to as research and development, or in educational contexts, “design-based research”. Translation is critical, however, because it is what allows designers to convert abstract relationships of forces into structurings that channel the forces to where they can produce an intended effect.

In addition to the science-technology theory dichotomy, there exists a possibility of distinguishing at least two varieties of technological theory important to this study: (1) design theory (Simon, 1999), and (2) domain theory (Schon, 1987):

- *Design theory* is concerned with the manner in which designs are made. It describes approaches to design problem solving that can have direct influence on the designer's choice of activities (Simon, 1999). Design theory tends to be portable across design disciplines. Gibbons' theory of layers is an adaptation of layer theory from many fields, to create a design theory for instructional design.
- *Domain theory* guides the selection and arrangement of design elements within a particular layer of a design (see Schon, 1987). Domain theory is usually framed within a particular design field. In some cases, domain theories are portable between fields, but in general theories of electrical grid design do not apply to the design of instruction. Domain theories tend to be related mainly to the design questions of a limited domain of artifacts and the concerns of a particular field of design and its immediate neighbors.

This study surveys portable theories of control system design from a number of fields as they apply to the design of educational experiences.

The Control Layer

The control layer is one of seven layers named by Gibbons (2014) in an attempt to define an approach to learning experience design based on the functionality of the artifacts being designed. A layer is a division of a design problem determined by the designer to have semi-independence of other parts of the overall design problem. Layers are an individual choice of the designer. The seven layers of an instructional design suggested by Gibbons are divided and subdivided into sub-layers that focus the designer's attention on specific areas of the design, while at the same time allowing individual layers to be integrated and harmonized with each other. Layers serve purposes of division of labor, product modularization. Most important to this study, layers aid in identifying and organizing bodies of theory that afford design principles to instructional designers. Layers make it possible for the designer to see entry points through which theoretical principles can influence a design.

The control layer was selected for this study because it is less obvious to most instructional designers. Few would suspect that it would be a fit subject for theoretical study. Control systems are an afterthought in many instructional designs. We originally felt that if we could show that the neglected control layer possessed a relevant body of theory, then we might use the control layer as a pattern for illustrating theory-design linkages for the remaining layers as well.

In its normal function, the control layer is one of three semi-independent layers that together define the "interface" of an instructional design. Besides the control layer, the other layers that participate in creating the interface are the Message layer and the Representation layer (see Gibbons, 2014). Functioning together, these three layers supply the means for a two-way conversation between a learning source and a learner. Though the layers of a design function together, the design requirements, design questions, and design theories for each layer differ considerably.

In control layer design, a designer anticipates the range of contexts in which the learner will be placed and the range of possible learner intentions that need expression within those contexts. Control system design therefore consists of providing tools for the expression of learner intentions within designed learning contexts.

The control layer entails the design of learner-operated expressive controls. Since WW II, control system design has grown rapidly as an area of the study within human-machine interface design. Vincenti (1990) provides an extensive case study of the growth of control design theory in aeronautics, noting that one of the challenges was to notice the problem in the first place: "The engineering community did not know at the beginning of our period [of 25+ years] what flying qualities were needed by pilots or how they could be

specified” (p. 51). Describing virtuous control qualities was a problem that had to be solved over the course of more than two decades. As more sophisticated aircraft designs emerged, new aspects of the problem became apparent. As the sophistication of instructional designs increases due to new competitive pressures, we wonder whether a similar period of discovery will be required to define virtuous control qualities for instructional designs.

Instructional Control Systems

Instructional control systems seldom receive attention in the educational technology literature in their own right. Therefore, this section describes a selection of control system types under five categories of control function.

Administrative Controls

Administrative controls are used in contexts free of specific subject-matter. Administrative controls launch programs and services, provide administrative communication, and supply help. Administrative controls are found on the representational surfaces of operating systems, application software, and web browsers. Administrative interfaces normally provide so many potential actions that some controls are hidden within menu hierarchies. Administrative controls take the form of icons used to run applications, folders that can be opened, and option lists on the control panel.

In addition, for instructional designers, administrative controls perform actions independent of specific subject matter. Therefore, administrative controls are important to learning management systems. Figure 1 illustrates the TREKKER system, a set of controls for launching instructional events and making administrative decisions. Using such controls, the user is like a trekker, who has emerged from the woods into a clearing to scan the terrain and chart a path. Context-sensitive TREKKER controls can offer appropriate choices at key transition points--between instructional events or even within instructional events.

Transitioning Between Instructional Events

Seek Orientation	Evaluate Status	Form Goals	Plan Actions	Provide Info
Seek Orientation	Evaluate Status	Form Goals	Plan Actions	

Transitioning Within Instructional Events

Provide Info	Seek Orientation	Evaluate Status	Form Goals	Plan Actions
	Seek Orientation	Evaluate Status	Form Goals	Plan Actions

Figure 1. TREKKER administrative controls (From Gibbons, 1997)

In both between-event and within-event contexts there are controls for: (1) seeking orientation, (2) evaluating current status, (3) forming and expressing goals, and (4) expressing action plans. The crossing of the two contexts with the four functions creates eight separate contexts for the provision of controls (and related decision-supporting data).

Navigation Controls

Navigation controls are associated with the exploration of information and action spaces. They afford movement within the space, facilitate inspection of the contents of the space, and bring the user to places where action is possible. Google Earth, the Authorware camera demonstration, and resource searching using a search engine provide examples of this type of control.

Google Earth is a very large set of controls for examining geographic surfaces and their related databases. Google Earth controls permit the user to navigate to any point on a natural surface using a variety of commands. Controls function with pinpoint accuracy to take the user to a set of coordinates or to a physical feature. Controls are also provided to reveal overlay markings representing roads, political boundaries, named places, advertised places, photograph placements, and other data items.

The Google Earth controls permit the aspect and range of the user's view to be changed. Other controls allow seamless surveying of a moving terrain, the deposition of comments and visuals, the replay of comments and visuals, and the capture of traces in the form of fly-buys.

The control set of Google Earth has been abstracted and fitted to multiple similar products. It has become a product and a metaphor in its own right through application to such applications as Google Moon, Google Mars, and Google sky.

The Authorware camera demonstration (used as a marketing tool for Authorware, a 1980s development system that is no longer supported) provides the basis for another example of navigation—this time by exemplifying what is *not* there. The Authorware camera demonstration originated in the 1980s as an authoring demonstration for a then popular authoring system. Using the demonstration, a user could set the F-stop, shutter speed, and film ISO value on a camera and then press a shutter control on the display (see Figure 2). A snapshot depicting the results of the user's three settings would then appear in a small window at the bottom of the display. From this the user could determine the effects of the settings on the appearance of the snapshot (dark, washed out, etc.). After viewing this image, the user could then change the settings and snap another picture, viewing the results in a new image displayed in the same window. This process could be repeated as many times as the user wished, each time producing the same picture, conditioned by the new settings.

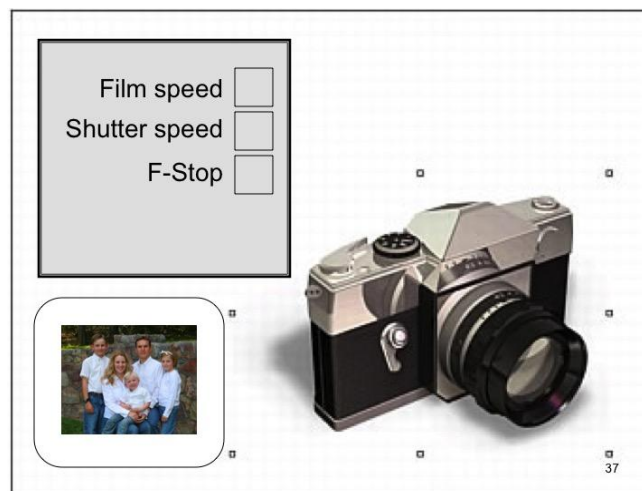


Figure 2. Re-creation of the Authorware camera demonstration display.

The demonstration was a powerful marketing tool, but it fell short instructionally. The fault lay in the inability of the user to compare snapshots. Each time a picture was taken, the previous picture in the viewing window was overwritten. The view space theoretically being navigated by the missing control set is illustrated in Figure 3 as if it had existed. Additional value could have been supplied by the ability to compare *changes* in the picture value from the previous picture. Multiple contrasts representing different settings would

eventually create a matrix of contrasting pictures along three dimensions. Giving the user the ability to navigate trajectories within this matrix would allow the user to view those contrasts and study the combined effects of different settings of the three controls. Not only were there controls for setting picture variables, but there could/should also have been controls for navigating around the space of pictures taken with different settings combinations. Thus, one function of navigation controls can be to allow the learner to explore spaces looking for differences or contrasts, rather than objects.

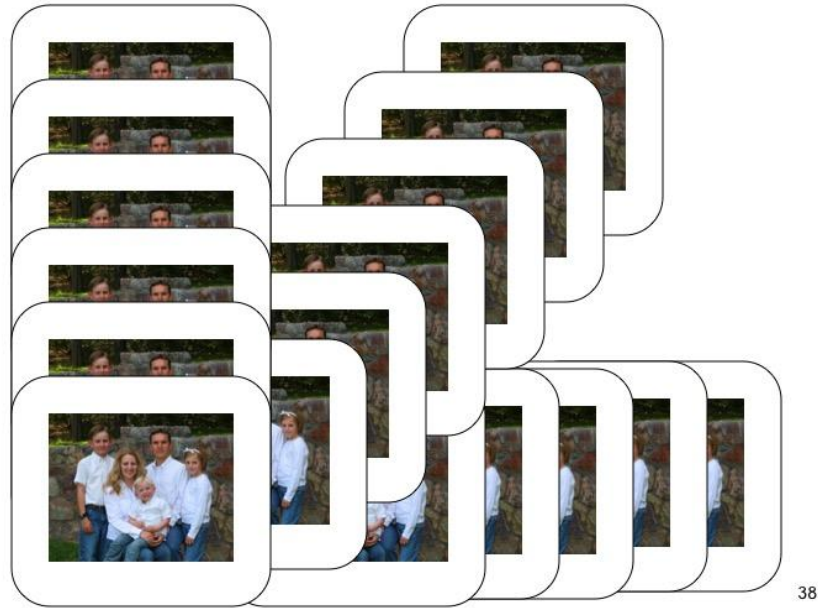


Figure 3. The representation space navigated by the missing Authorware demonstration controls.

An additional control set of this type is demonstrated by Hans Rosling using software called Gapminder (See Figure 4). The principle of Gapminder is that a multivariate database is visualized dynamically in which one dimension represents change over time. When Rosling presses the Play control, the changes over time become represented as movements of graphical objects representing points within the data space. In a sense, Gapminder adds dynamism to the control-driven exploration of a data space. A control system used for setting range values and naming data vectors is used to set display parameters before the Play control is even pressed. Once play is initiated, graphical representations of the data are created according to the control settings. These controls can be categorized as navigation controls, because each setting made controls the unfolding of a motion-dynamic data space that the user can then observe. Changes in control settings change the unfolding and therefore the information available within the representation of the space.

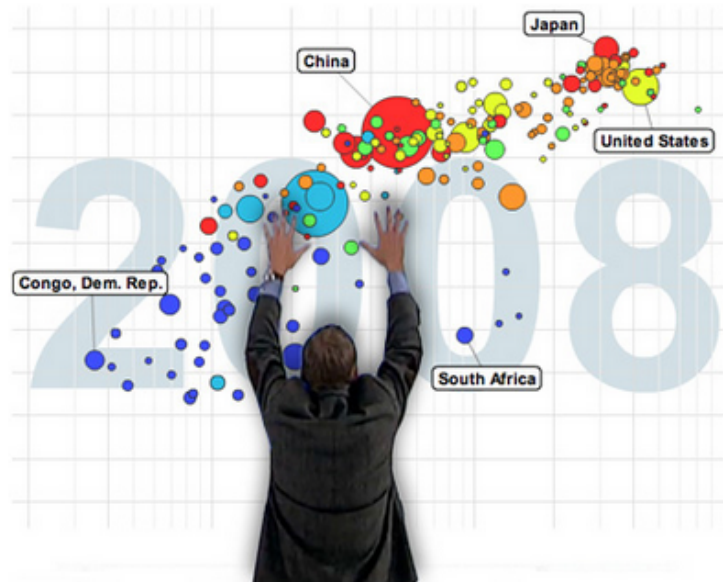


Figure 4. Gapminder's dynamic display of database values. (from <http://www.gapminder.org/>)

Web links are also examples of navigational controls. We commonly refer to "navigating" the web. One means of navigating is to chain from Web page to Web page, along an unplanned path over which we "drill down" across several linked pages.

Action Controls

Action controls are associated with action contexts—for example, where learners are operating or controlling the processes of a system or a device for experimental or procedural purposes. The most common examples of action controls are found within simulations, which involve performing actions on dynamic, computed models of cause-effect systems. Most game controls are included within this category. Action controls are often associated with continuous modeling processes that require regulation, adjustment, and ongoing attention. The operation of action controls usually takes place within a context where tracking, difference detection, and cybernetic course adjustment are involved.

The context of action control operation is one in which a learner makes a comparison of a prior momentary state with the current state and acts to bring about a desired future state. Tracking and responding to dynamic model state changes are key concepts for this kind of control. The model processes controlled may be either single-cycle or continuously-cycling. Therefore, the controls themselves may be operated in either discrete or continuous modes, depending on the timing of the simulated process.

Statement and Query Controls

Statement query controls are associated with contexts where verbal or symbolic expressions are exchanged. Control operations take the form of a semantically significant entry: either a question or an answer to a question. Since symbolic actions are included in this group of controls, raising a hand in a class setting constitutes a control in this sense. Statement and query controls allow the kinds of expressions that make up a conversational exchange.

Statement and Query controls can have greater semantic range than other types of control. Therefore, may consist of commenting, annotating, responding, composing, seeking, creating, structuring, hypothesizing, requesting, suggesting, explaining, asking, sketching, outlining, arguing, requesting, claiming, giving evidence,

countering, relating, asserting, disputing, countering, illustrating, proposing, exemplifying, connecting, imagining, persuading, agreeing, supporting, or concluding--all expressive or inquiring acts of a conversation.

Statement and query controls are commonly found in homework assignments, class exercises, discussions, quizzes, and tests. They are the most common form of classroom control exercised by a learner during instructional conversations. Achieving a more robust implementation of statement and query controls using technological devices has been the preoccupation of natural language processing research for many decades. The fruit of this processing research is now appearing on "smart" phones everywhere. Though the semantic processing of spoken and typed controls is less than perfect, lexical and syntactic processing has improved, and natural language interpretation tools are found on almost every computer and in search engines. Search engine controls were also mentioned in connection with the navigation category. This demonstrates that multiple categories of control may be afforded to the user at any given moment during the use of an application and also that these categories blur at the edges.

A research project in the 1980s called CSILE (Computer-Supported Intentional Learning Environments) (Scardamalia, 2004), which eventually evolved into the Knowledge Forum(c) commercial product, exemplifies a technology-based instructional interface that makes non-trivial use of statement and query controls.

CSILE is a virtual collaborative problem-solving space. In the CSILE space learners conduct personal projects, while at the same time reviewing and critiquing the projects of other learners using the same virtual space. Interactions related to projects can consist of simple commentaries, engaged debate, or collaborative research as a team. In terms of user control, interactions between users are carried out through a structured conversation in which only a handful of common expression types are legitimized as what are called "scaffolds" that discipline comment types.

Scaffolds, which are preselected by the space designer, may include "propose a theory", "supply evidence", "supply counter evidence", "propose an explanation", "hypothesize", "propose an experiment", and others. Many kinds of projects can be carried out in a CSILE space, including debates, joint experiments, text composition, and group creative work. Scaffolds chosen to structure communications are in essence meta-controls: controls that are composed of more elemental controls. They discipline the conversation, and over time they become internalized as types that can be used outside of the project space. Statement and query controls used during conversational interactions can resemble action controls, especially when expressions consist of arrangements of symbols, icons, or manipulatives.

An example of this is provided by the Osmo(c) interactive learning games (see Figure 5). An Osmo mirror redirects an iPad camera's field of view downward. When the iPad is placed upright in a base, the redirected camera turns a desktop into a work surface on which objects can be arranged. Osmo software can identify the shape, arrangement, and color of objects on the work surface. It compares these against anticipated arrangements. When a match is detected (e.g., letters in a target word or a specific arrangement of tangram shapes), pleasant sounds and sights signal success to the user, and points are awarded.

The Osmo system combines Action controls (arranging the symbols or objects on the work surface) with Statement and Inquiry controls in a unique way, because actions are used to form expressions that are in some way grammatical and semantically meaningful. Using manipulatives as controls, the user creates expressions, and then the computer responds with a judgment of what it sees.



Figure 5. The OSMO interface, in which the controls are manipulatives detected by a downward-looking camera.

Oviatt (2013) describes a class of controls that employs an electronic pen capable of writing notes, marks, and diagrams on the display surface in a capturable way. These can be considered as linguistic and symbolic expressions. Osmo provides this kind of control also in a game called Newton, where written marks on a sheet of paper are interpreted and processed as barriers. Bouncing pellets fall from the top of the Osmo display until they hit a barrier, which causes them to bounce away. The goal of the Newton game is to draw barriers on paper that bounce the pellets on the iPad display into targets that Osmo has placed. The remarkable feature of this control system is that the control emerges from the user's choices and actions in real time. There is no barrier until the user draws a line or a curve on the paper within the camera's view.

Negotiation Controls

Negotiation controls are associated with contexts in which *commitments* are being made and fulfilled. This normally involves the process of setting and monitoring the attainment of learner goals or the adjustment of some aspect of the learning strategy. Negotiation controls are a specialized subset of statement and query controls. Their signature feature is the persistence of the commitment made, which allows periodic monitoring of accomplishment. Negotiation controls are formative conversations in which the agency of the user and the agency of an instructional source come to a mutual agreement. What is being negotiated is a cooperative contract or compact between the learner and the learning source that will require the efforts of both parties for fulfillment.

Qualitative Dimensions of Control Systems

Control systems are central elements of virtually every human-machine system, of which instructional systems are an example. Industrial applications of control systems are too numerous to catalog. Perhaps the best metaphorical instance of a controlled system is the computer.

Cybernetics supplies the principal theory base for controlled systems. A cybernetic system influences its surroundings but also senses the environment and modifies its behavior on the basis of what it senses. During an instructional conversation, the cybernetic elements include a responsive source of learning experience (instructor, mechanism, etc.) and one or more learners. As learners act, the learning source senses and responds according to whatever fixed program or intelligence exists in the source.

Conversational instructional interaction requires some means for the learner(s) to express action, which implies a control system. Control systems, whether simple or complex, possess properties that associate designs with theoretical principles. Many of these properties pertain to control systems in any field of design: an aircraft cockpit, an automobile interior, a heavy equipment operator station, or a workplace environment. Ergonomic studies list in great detail the qualities related to control systems with acceptable properties. The list below is representative of a pooling of qualitative concerns from a variety of design fields. (See, for example, Yeh et al, 2013 (aviation); International Organization for Standardization, ISO 4040:2009 (2014) (automotive); Green et al, 1994 (automotive navigation); American Society for Testing and Materials, 2013 (medical equipment). The list below, which is abstracted from standards similar to those referred to above, suggests a range of qualities designers of control systems should consider as they design.

Transparency

Control systems should not call attention to themselves, except to make their affordances apparent to the user. Control systems should minimize cognitive load over and above the processing required by the target performance. A control system should avoid the requirement for special training in control use. A control system should allow the user to maintain an adequate level of concentration on the target task. (To promote transparency, a control system design should be accompanied by a messaging and representation design capable of supplying system data necessary for operation.)

Efficiency/Responsiveness

Control systems should respond rapidly to changes in user input. They should synchronize closely with changes in the representation of indications. Control actions should require the minimum number of operations. Controls should be operable with a speed and accuracy appropriate to performance criteria. Operations on controls should be accompanied by confirmation that operations have been accepted and processed.

Affordance

Control systems should offer the user the ability to carry out actions appropriate to the current performance goal. Controls should be salient in proportion to their importance to the performance and/or the urgency of their use. Controls should be readily distinguishable from each other. Controls that look similar should be separated sufficiently to prevent unintended operation. Controls should be designed to suggest the action they perform.

Fidelity

Control systems should look, feel, and act naturally if they are used in simulations. Control state should be clearly visible. Controls operated upon should display their new, changed state if there is one. Control feedback should be continuous if the operation itself requires continuous action. Response to controls should be consistently in proportion to the degree of control operation.

Anticipation

A control system may anticipate an action to afford the user if it helps the control system to be more transparent. Control design should take advantage as much as possible of existing conventions (e.g., QWERTY keyboard layout, habitual patterns of use).

Underlying Model

A control system should be based on a single (mental) model of the operated system that the user is expected to possess. A control system should be styled according to a consistent metaphorical, semantic, or stylistic “theory” of the control system.

Consistency/Unambiguity

Controls should operate in the same way each time they are afforded to the user, producing the same result when used in the same context. Voice activated controls should produce the same interpretation of commands over repeated use. Voice interpretation should be unambiguous and accurate.

Depth/Hierarchy

A control system that is hierarchical should reveal new control options in an order consistent with the user’s model of the system being operated. A control system should not include hidden functions that require special, non-intuitive operations to make them visible. A control system that employs modes of operation should avoid the possibility of becoming trapped within a mode.

Precision

A control system should be matched in precision with the data available to the user during operation. A control system’s response should be precise across the entire range of action.

Expressiveness

A control system should provide a degree of expressiveness that is consistent with the criterion of the performance being measured. A control system should allow a user to express all appropriate intentions to the system. A control system should be aware of the context of control action if that is necessary to interpretation of the user’s intention.

Accessibility

Controls should be made accessible to all users within the target demographic. A control system should be designed for ease of use in the target environment (including lighting, posture, ease of movement, etc.). All parts of a control system should be designed with a defined model of the user and user capabilities in mind.

Safety

A control system should not lead to injury to the user. A control system should not create a surprising or startling distraction from the task being performed. A control system’s operational range should be limited to within safe boundaries to the extent possible. Control systems should be designed to reduce user fatigue.

Recovery

A control system should allow a user to recover from mistaken actions.

Summary

This list of control system properties taken from a variety of control design standards is not exhaustive, but it suggests an unexpectedly broad range of practical principles that designers must observe to make control systems usable and useful. These properties apply to instructional designs. The development of control

design principles over time has been the result of either post-incident error analysis or proactive research and development involving some combination of participatory design, prototyping, and/or user-testing.

These principles taken as a whole are the product of human-machine studies that accelerated with the increasing complexity of systems between the first and second world wars. During this long history, control design principles have emerged from two sources: (a) practical considerations, and (b) application of theory. It is appropriate, therefore, to consider these properties as entry points for theory into control system designs.

Examining the Theoretical Bases of Control System Design

Instructional control systems are designed to support the exercise of critical mental processes that will lead to an increased probability of learning. We assume that control design can be informed by theory. Understanding theory allows the designer to translate conceptual structures into real structures that preserve the energy of the designer's intent. This section attempts to relate several control-related theories to principles for control design.

We examine theory-design relationships at multiple levels. First, we consider the traditional theoretical divisions of behaviorism, cognitivism, and constructivism for their implications for control design. Second, we consider a number of other theories originating in diverse fields.

Behaviorism, Cognitivism, Constructivism

Three schools of learning and instructional theory are frequently taught in instructional design programs: behaviorism, cognitivism, and constructivism (see, for example, Lowyck, 2014; Ertmer & Newby, 1993). These divisions of theory differ in their concern for control design principles.

Behaviorism

Skinnerian behaviorism had great impact in the 1960's on the formative field of instructional design, producing research on teaching machines: an emphasis that later morphed into research on machine-less programmed instruction (Benjamin, 1988). The varied forms of programmed instruction possessed in common a requirement for frequent responding on the part of the learner, usually within the context of a "frame" (Markle, 1969). Styles of programmed instruction were characterized by different definitions of "frame" and different modes of responding.

Teaching machine controls were physical and provided numbered or lettered buttons, levers, keyboards, touchpads, and/or small openings for written answers. Paper programs used either masking cards or page separation to hide correct answers while the student responded.

The physical control mechanisms of teaching machines turned out to be non-essential. This was one of the main reasons for the disappearance of the machine as a responding device. It was apparent that a program's effect was created by the requirement for constant responding, even if in some cases it was covert. Control design remained an integral element of program design, because other mechanisms for hiding and then revealing the correct response were still used. Control design was not treated at the time as a topic in its own right in the literature, other than as a necessary characteristic of responding.

A robust culture of research on programming probed many possible configurations of frames and frame sequences. Variables included size of frame (internally), and step size between frames (pace), degree of prompting, rate of prompt fading, type of feedback, acceptable error rate, error correction logic, and others. In 1967, at the peak of interest in programmed instruction, Anderson (1967) observed that, "it is now widely believed that the behavioral science must be engineered into teaching methods and materials" (Anderson, p. 129). By that he meant manipulation of frame properties, which included the type and rate of responding.

Design at the time was empirically based, involving goal identification, behavior/subskill analysis, “and as many cycles of tryout, reanalysis, and revision of the lessons as are necessary to attain the objectives” (Anderson, p. 129). Design was not at the time considered a predictive theoretical exercise as much as an empirical one. Susan Markle, Skinner’s protege, strongly defended this empirical approach to program design (Markle, 1967).

The common thread running through programmed instruction designs was a continual cycle of responding, followed by feedback. The differences were mainly the length of time and/or the amount and type of information the learner experienced before the opportunity was presented for responding. We estimate that control design was in fact as important in programmed instruction design as the principle of reinforcement itself, since control operation, whatever form it took, was the main mechanism for delivery of the reinforcement. In many forms of programmed instruction, it might be questioned whether true reinforcement was occurring, because of the lack of a deprivation state in the learner. It cannot be questioned, however, that exercising control was a consistent design factor in instructional programs.

Cognitivism

Behaviorism manipulated variables external to the subject’s mind; cognitivism concentrated on the study of mechanisms (information processing) and knowledge representations (models, schemata, etc.) within the mind. Externalized control operations and responding in small increments became much less important. Mandler (2002) describes the “cognitive revolution” that took place over a period of roughly 10-15 years, from about 1955 to about 1965. Instructional designers who lived through this revolution saw behaviorist terms such as “stimulus”, “elicit”, “feedback”, “responding”, and “reinforcement” largely disappear from psychological literature. Wilson, Jonassen, and Cole (1993) describe the implications of this shift for the instructional designer: instruction became more job-like, instructional goals became more complex and high-end, and the concept of learning environments gained traction. Progressions of learning challenges and environments became a subject of study. A body of literature grew describing cognitive task analysis methods. Learner goal processes and knowledge states became a feature of adaptive, intelligent tutors, and artificial intelligence and expert systems appeared in the designer’s toolbox. As the description of the learned content grew in complexity, so did the complexity of assessments and the judgment rubrics used to determine competence.

The concerns of the instructional designer, therefore, changed from frames and frequent responding to creating learning environments (e.g., simulations, microworlds, collaborative spaces) for problem solving and/or exploration. Assessments focused on higher-order, lengthier, and complex performances. Verbal responding methods were subordinated to actual performance where possible.

Attention shifted from empirically-based to principle- and theory-based design. Controls in the new instructional setting became more expressions of learner goals and navigational intentions within problem environments and less correct responses to stimuli. Direct instructional methods were joined by new methods that enlisted the learner in solving, discovering, and designing. Modes of control expression like navigation and negotiation became more common, allowing learners to operate within choice- and initiative-rich learning environments.

Equally important, the meaningfulness of individual control operations changed, and the semantic, context-dependent dimension of control operation became more important. Operations of controls became meaningful expressions subject to some degree of interpretation to determine the learner’s intent at the moment of use.

Constructivism

The educational philosophy of constructivism maintains that different people construct their own realities. Within education, constructivists view learners as constructing knowledge; they tend to see instruction as a process of supporting knowledge construction by the learner (see Cunningham & Duffy, 1996). This

philosophy has promulgated a broad-range of instructional models that emphasize scaffolding, ill-structured problems, feedback, collaborative learning environments, and knowledge-building (Scardamalia & Bereiter, 1994).

Social learning theory merged with cognitivism, introducing new choices for distributing roles and responsibilities among learners, instructors, and peers (Lave & Wenger, 1991). This led to the principle that initiative could be shared dynamically among participants. The theory of cognitive apprenticeship (Collins, Brown & Newman, 1989) is an expression of constructivism, joined with social learning theory.

In the same way cognitivism reduced the demand for external and frequent responding compared to behaviorism, constructivism reduced attention to the nature and form of control operations compared to cognitivism. With each change of worldview on learning, greater emphasis was placed on agentic learning, self-regulation, learning within communities, and the instructor/designer in the role of a coach and mentor rather than a determiner, chooser, and judge.

The reduced attention to the specifics of control operations by the learner at each step did not remove the designer's responsibility for selecting appropriate kinds and amounts of control action, relative to the goal of learning. However, guidelines for designers of control systems were left a matter of individual preference, and systematic attention to the principles of control design disappeared as a topic for most instructional designers. Only in the field of simulator design did control design continue to exist and grow. Consequently, the development of control theory concepts has become largely neglected as a topic in the literature.

Specialized Theories of Control Design

Over time, the traditional theory types reviewed above migrated away from making statements regarding control design. As a result, there is little literature giving guidance to the instructional designer, who nonetheless must still design control systems. The sections that follow describe several local theories, mostly from outside of educational technology, that do contribute to control system designs in significant ways. However, an important distinction must first be drawn between the design of control systems and the design of interfaces.

Interface Design

We feel there is an important distinction between what is generally referred to as "interface" design and what we are here calling "control system" design. The concept of an interface is more inclusive; a control system is but one part of an interface.

Interface design literature addresses the totality of an experience generated by: (1) a representation (source of sensory experiences: visual, auditory, kinesthetic), (2) a set of controls for user responding, and (3) messages that drive the content of the representation. The division of the interface design into these three functional parts is proposed by Gibbons (2014), with each functional area of the interface being considered a "layer" of the interface design. This division encourages the designer to see and respond to more detailed design questions whose answers draw upon different bodies of theory. The message layer tends to derive principles from conversation theory; the representation layer is to theories of symbolic communication, including both psychological theories and semiotics; the control layer draws upon control system design theory.

Cybernetic theory informs and unifies the design of interfaces as a whole. Cybernetics is the study of feedforward (anticipatory) and feedback (responsive) systems. In both of these cases, a system and its environment interact in a "conversational" manner, each influencing the other. Some of the control design theories reviewed later in this section involve cybernetic principles, but the emphasis of the reviews will be on the contribution of that part of the theory related to control design.

Theories for Control System Design

Theories related to control system design can be found in the literature of developmental psychology, educational technology, computer science, game design, ergonomics, and computer interface design.

Papert and Resnick: Constructionism

Constructionism (Papert, 1980; Resnick, 1997) is a theory of learning by action-taking based on constructivist principles. Action-taking provides learners the means for experimenting: hypothesizing, making, executing, and observing results. Performed multiple times, this experimental cycle allows a learner to create personal “theories” that turn into knowledge.

The psychological core for constructionist ideals is traced to Piaget, but many additional theoretical contributions inform constructionism in its current state. These include conceptual additions embodied into practical innovations, usually in the form of computer programming systems. Key embodied systems include the Logo (McNerney, 2004) and Scratch (Resnick et al, 2009) programming languages (and their interfaces), and Lego Mindstorms created in cooperation with the Lego Corporation as a commercial product (Resnick et al, 1988; LEGO Mindstorms, n.d.).

Support for action-taking is the key principle underlying constructionism, and many kinds of occasion have been devised to motivate action, virtually all of them based in some form of constructive play. Learners use constructionist environments to design, invent, discover, and create.

A theory of control design is embodied by the Scratch programming language and its interface. A small number of color-coded graphical “programming blocks” (representing executable program actions) are dragged into sequences in a growing program within a workbench-like visual space. The shapes of connectors on the blocks suggest legal connections. The current control design theory for Scratch includes making it “tinkerable” (users can start building right away at an intuitive interface), meaningful (permitting diverse projects, easy-to-create and personalized content), and social (encouraging joint projects, collaboration, community sharing, and remixing). All of these qualities of the language and interface are centered on basic principles for allowing the learner to take action.

The Scratch interface is the forward projection of years of experience with predecessor object-oriented and visual programming systems like the LOGO and SmallTalk (ref) programming languages. These predecessors were experiments in making computer programming easier and more approachable to a less technical audience. The principles of most importance, refined over time through these experiments, include controls that “offer” themselves and scaffold their own use through shapes, colors, and simplicity. Visible shapes constitute terms in the programming language but also double as controls that are operated both by moving them into place and by executing their actions.

Limiting the number of program commands (controls) in order to maintain simplicity has been a challenge. The basic palette of Scratch controls introduces basic programming concepts, but these are only the beginning of more powerful but abstract concepts that could be embodied in controls and offered to the user (Maloney et al, 2010). Multiplying controls that operate at different levels of abstraction and complexity is a problem that the Scratch development team will solve as they invent new theoretic principles of control design.

Wertsch: Mediated Action

The theory of *mediated action* is relevant to the study of control system design because it is a theory of how individuals act on and influence their environment. James Wertsch (1998) studies mental function in terms of mediated action: action by an agent that employs an instrumentality or tool.

Wertsch studies the “agent-operating-with-mediational-means” (p. 24), which implies that his studies are directly relevant to the design of the means themselves, including their visible manifestations and their behavior. In Wertsch’s view, mediational means (tools) include not only objects, like the pole used by the pole

vaulter, but ideas, traditions, and meanings passed to the individual through common use by others who take part the same cultural milieu.

Language is an example of a non-object mediational means. People use language to accomplish communicative actions within a cultural environment. Language is defined by a culture, but the actions of an individual can also impact the language. Therefore, there exists a two-way influence between the mediational means and the agent who uses it.

Wertsch makes ten basic claims that “characterize mediated action” (p. 25), of which the eight considered below have direct implications for control system design:

1. **That mediated action typically has multiple simultaneous goals.** The student may be trying to maximize a grade, while at the same time trying to discover shortcuts that allow more rapid completion of a task. The need for shortcuts may indicate a situation that Wertsch describes where “the goals of the agent do not map neatly onto the goals with which the mediational means are typically associated” (p. 34). Control design, by implication, should understand the range of user goals within a fluid context of learning and anticipate user initiatives.
2. **That mediated action is situated on one or more development paths.** As users employ mediational means, both change: “both agents and tools are always in the process of undergoing change” (p. 34). A tool that was sufficient for the user as a novice may hamper expression of action as the user becomes more competent. Therefore, a control system designer must take into account maturation of the user and provide a way for escalating competence to be expressed, as well as entry-level lack of competence. A clear principle emerges that the design of a tool defines the range of development it can support in the agent who uses it.
3. **That mediational means constrain as well as enable action.** Every tool facilitates one set of actions while at the same time making others more difficult. The control system we know as the QWERTY keyboard was designed deliberately to slow down typing to a speed that early typewriter mechanisms could support. Later invention of the Dvorak keyboard made faster typing speeds possible, as shown in typing competitions. For one-handed typists, special “chorded” keyboards have been developed that solve constraints imposed by both the QWERTY and the Dvorak keyboards.
4. **That new mediational means transform mediated action.** Changes in cultural tools transform patterns of action, sometimes modifying the actions themselves. New operating system technologies changed the way users executed basic computer functions; a new tool called the browser transformed the way networked computers could be used. The inverted mouse that we call a trackball introduced new navigational possibilities that did not require arm movement. Google Earth introduced navigational tools that facilitate exploration of a spherical surface. This tool proved portable to products like Google Moon and Google Mars. Some elements of this control system are also used in Google Maps. Wertsch quotes Vygotsky’s observation that “by being included in the process of behavior, the psychological tool...alters the flow and structure of mental operations” (p. 43).
5. **That agents “master” mediational means.** It is a temptation to think of control systems as being “internalized” to the point of automatization, but Wertsch offers an additional concept of mastery, which creates the distinction that some operations on tools may be mastered without becoming automatized, and this may be deliberately so. Acts of driving and walking become automatic, but the skill of adding columns of numbers with carrying does not seem to mature in the same way. Some control systems may be designed to be only mastered, while others may be designed to become automatized.
6. **That agents “appropriate” mediational means.** Agents may become proficient in the use of mediational means (such as the ability to reason within an idea system) without becoming committed to it. Agents have agency by definition, and they may choose to resist fully-committing to certain tools, while accepting their use (e.g., Algebra). This may be exemplified by the selective use of smartphone functions. A user may accept the phone itself and use it to make calls while resisting

messaging or other social media functions. Likewise, an agent may wholeheartedly appropriate a mediational means, such as a second language, learning to speak it without an accent. The first author asked an Englishman if he could speak without the British accent. He said, without any trace of accent, "Yes, but it hurts my throat."

7. **That mediational means are not always created intentionally, nor are they necessarily used in the way intended.** Wertsch proposes that "most of the cultural tools we employ were not designed for the purposes to which they are being put" (p. 59). In Wertsch's writing about mediational means, there is a strong sense of the evolutionary nature of the means and of the role of serendipity in finding appropriate tools and adaptations of tools to new uses. This would seem to indicate that there are no "right" answers to tool or control design questions, but that experimentation might be used as an approach to arrive at satisfactory tool designs.
8. **That mediational means are associated with power and authority.** Power and authority in instructional designs are often expressed subtly, even in designs deliberately arranged to reverse traditional authority relationships. Wertsch proposes that the emergence of new tools can transform power and authority relationships. However, there is a trap in assuming that new tools completely reverse the balance of power. If instruction is seen as the cooperative interaction of two agents for a purpose, then an appropriate way to conceive of the role of mediational means (tools for the expression of choice) is to see it as granting appropriate levels of choice to each of the agents involved in the relationship. That is, controls become a critical point in the negotiation of levels of choice assigned to each agent and become a tool in a larger sense for drawing forth and educating the executive competencies of the learner with respect to their own instruction and learning.

Wertsch's theory describes the interaction of environment and the individual through tool use. Wertsch's principles can be readily translated into guidelines for control system design.

Crawford: Linguistics and Control Design

Game designer Chris Crawford is vitally interested in the practical aspects of full interaction design. However, in his book *The Art of Interactive Design* (Crawford, 2003) he outlines a theory of control design based in linguistic principles. Though his first concern is practical design, Crawford offers design advice at a more abstract level:

The breakneck pace of technology induces many designers to grab for quick solutions. But it is precisely the torrid pace of development that requires us to concentrate on understanding the fundamentals rather than snatching quickie patches. Today's solutions will be obsolete in a few years, but the fundamentals (and this book) will still be useful years from now. (From the preface, "Read Me")

Crawford's game interfaces must possess the qualities of transparency, agility, and expressiveness discussed earlier. These game control systems for him are not ancillary in the design: they are a central issue, because at their core games consist mainly of prolonged sequences of control operation (working in coordination with message and representation generation).

Crawford's philosophy of controls characterizes interactions as a conversation. He states, "At every step in the design, you must ask yourself, am I talking *at* my user or talking *with* him?" (p.29, emphasis added). Crawford names three activities he considers necessary for conversations: listening, thinking, and speaking. He notes that computers and humans speak different languages, with the following result:

You shout at it, point to the screen, stab at the keyboard with your fingers, and you know perfectly well that the computer sees you as a grunting, inarticulate deaf-mute. You know what you want, but you can't find the words to say it to the computer.... The failure arises not from your lack of communication skills, but rather from the deficiencies of the language offered by the designer (p. 49).

Crawford's conversational theory of control design uses linguistic terms: "you must give your user the language to speak well" (p. 49). However, he says, "the glorious advances in technology over the past 20 years have bypassed the listening side of our task" (p. 50). The computer he believes to be language-challenged in the expressive options it offers the user.

According to Crawford, helping the computer to "listen" to the user begins with expanding the user's expressive vocabulary, and "the input device determines the size of the vocabulary available to the user" (p. 51). He compares computer input means in terms of the vocabulary they offer to the user.

- Single button--Two terms: yes/no.
- Joystick--18 terms: 8 directions plus button plus no control.
- Keyboard--Text character entry: few symbols, arrow keys for directional control; key combinations for shortcuts for power users.
- Mouse, pen, and touch screen--Pixel-addressable location; one or more buttons for selections.
- Voice--Limited only by recognition and interpretation software.
- Specialized control devices--Combinations of multiple buttons, joysticks, and steering controls (e.g., game systems).

Crawford's categories are based only on physical devices. Another way to categorize controls is as being either context-insensitive (single button, keyboard, and voice), or context-sensitive (all of the rest). Context-sensitive controls use the combination of display location (defined by hotspots, individual pixels, or audible contexts) and simultaneous control activation (button press, touch, or utterance). Context-sensitive controls increase the vocabulary of the user, but they place on the designer additional responsibility for: (a) context definition, and (b) interpretation.

Context definition adds a semantic dimension to control design: the user's meaning of selecting a pixel or a hotspot cannot be determined without knowledge of what the user is experiencing at the moment of control actuation. This principle holds true for audible displays as well. To design a context-sensitive control, a designer must first or simultaneously design contexts, which serves as dynamic menus with virtually infinite choices.

Context-sensitive controls naturally impose a second responsibility on the designer: interpretation of control actuations and sequences of actuation. Since the operation of a control can take place against the backdrop of a (virtually) infinite menu, the intent of the control at any given display location and moment in time must be interpreted relative to a location-moment pair. This principle is clearly identifiable in moving, changing displays, but is also clearly in operation for static displays as well; locating a static pulldown menu item and then selecting it requires interpretation, as does highlighting display options using arrow keys and pressing enter.

A second dimension of interpretation is added by voice and text input, both of which require not only the interpretation of primary symbols (individual text characters or spoken words) but the interpretation of symbol sequences (blocks of text and multi-word utterances). The complex technologies of text and speech interpretation have advanced rapidly into the commercial product world over the past two decades, but anyone who is spoken to a smartphone "assistant" recently realizes that the technology has far to go to realize its enormous potential.

Crawford further applies linguistic patterns by recommending that a first step in control system design be identifying a set of "actions (verbs) [that] most directly express the user's goals" (p. 62).

If you are developing an educational program, what is the likely mindset of your user? What does she already know, and what do you desire to teach? Do not look at it from your point of view as the teacher; look at it from the student's point of view. Specifically, don't just organize lots of information into tidy little structures that make perfect sense to somebody who already understands everything. Try to imagine yourself ignorant; what questions would you ask? By what circuitous routes might

your curiosity lead you to the information? Remember that you can't teach anybody anything; you can only facilitate a person's own learning process. What actions, then, would a curious student need to take to learn the material? (p. 62-3).

Control system design for instructional purposes involves anticipating user goals in advance within the different contexts of action that will exist at any given moment. To avert control chaos, this means defining a core set of actions (verbs) and in addition a set of objects (nouns) that can be acted upon, as well as the contexts in which they are active.

"Do it" is an even more abstract form of the command; it requires that the verb in question be set up previously by grabbing the device [display symbol] or otherwise unambiguously specifying the "it" to be done (p. 65).

Crawford organizes "its" into three main groups: menus (and menu items), maps, and gizmos. These "it's" supply noun-objects that can be acted upon. Menus and their hierarchical structures are familiar to the average computer user. Maps are also familiar, especially for users of virtual navigation systems like Google Earth; however, filing systems that place like items within a hierarchical grouping (folder) are also a type of map. Gizmos include familiar checkboxes, buttons, and text entry fields, but they also include hypertext links.

Crawford's linguistic approach to control design theory is home-grown. Crawford is a game designer whose experience, viewpoint, and credentials are clearly established in the real world of production and play. His theory is built from the ground-up experience as well as top-down imposition of a linguistic metaphor. It is a theory built through abstraction: "as systems grow bigger and more complex, they evolve more abstract structures to cope with the increasing complexity" (p. 250). Crawford's theory is an abstraction from his own individual game control designs, through multiple games and versions of games, that has found a useful and meaningful expression and linguistic terms.

Winograd and Flores: Conversational for Action

Winograd and Flores (1987) deal directly with control system design issues in a study of sociocultural influences on human-machine interface design. In their view, "we must take social activity as the ultimate foundation of intelligibility [between humans and computers]" (p. 33). Winograd and Flores question traditional categories of interface design, levelling existing assumptions to construct a new view of interface based on the philosophies of Gadamer, Heidegger, Maturana, and others.

Much of our theory is a theory of language, and our understanding of the computer centers on the role it will play in mediating and facilitating linguistic action as the essential human activity (p. 7).

Activity implies the use of different controls to carry out intentions. The operation of controls, according to Winograd and Flores, depends on the "concernful [goal-related] activity" we think we are engaged in.

Instead of trying to rationalize how humans should interact with computers, Winograd and Flores begin with the case of a typical computer user: normally someone with little understanding of its internal mechanisms. The focus of the user is on operating the software and not on how the computer and the software work. Winograd and Flores say that the computer user is "thrown" into the situation of operating the software to accomplish a purpose.

The user operates the software in pursuit of this goal until a breakdown occurs--perhaps some unexplainable program behavior. When this happens, Winograd and Flores propose that the goals of the user change, and the user is "thrown" into a new situation that requires new activity, new ways of acting, and new understanding of the software's operations: implying new controls as well. It is the user's goals and intentions rather than the technical design of the computer system that define what is needed at this point. Winograd and Flores contend that this makes the designer answerable to the user's functions and intentions rather than to the computer's (p. 39).

[To achieve his/her goals] the person selects among basic mechanisms, that the machine provides, to get work done. If the mechanisms don't do what is needed, others may have to be added. They will often be used in ways that were not anticipated in their design (p. 53).

When users operate controls to achieve goals in ways not originally intended, it is a clue that a rationalized design missed something: there is a user intention for which no means of expression was provided. That may be a sign that the designer failed to think of the interaction in conversational terms and the user as an agent with intentions that need to be carried out.

To answer this need, Winograd and Flores appeal to a theory of speech acts that considers the human and the computer as actors in a joint dialogue directed at the goals and intentions of both. The authors propose that "meaning is created by an active listening in which linguistic form triggers interpretation [and a corresponding speech act]" (p. 57). The speech act of one conversant results in a corresponding interpretation and speech act from the other. Speech acts constitute some form of commitment, even if it is nothing more than a commitment to continue interacting. Therefore, Winograd and Flores propose, "every language act has consequences for the participants, leading to other immediate actions and to commitments for future action" (P. 59).

On this basis Winograd and Flores propose a "basic conversation for action" form of interaction, depicted in Figure 6. This form consists of "requests" and "commissives" directed toward "explicit cooperative action" (p.64). This pattern, according to Winograd and Flores, can function "as the basis for computer tools for conducting conversations".

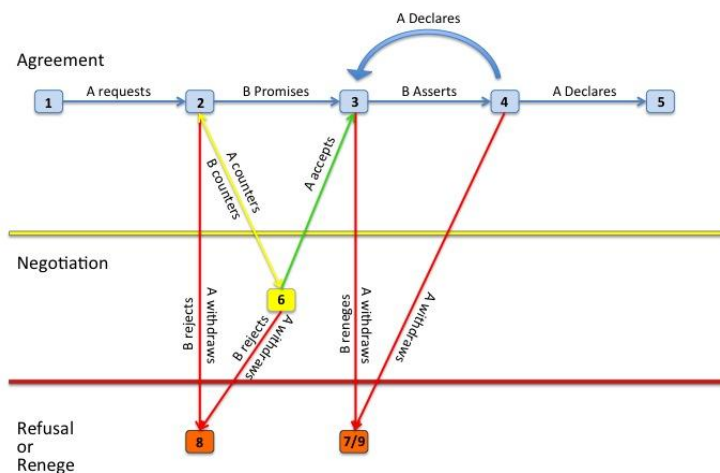


Figure 6. Winograd and Flores' basic conversation for action form (from Gibbons, 2014, after Winograd & Flores, 1987).

In Figure 6, numbered boxes represent states in a conversation for action. States are reached following speech acts--of five types identified by Austin (1962): assertives, directives, commissives, expressives, and declarations. Movement to the right in this form generally signifies agreement and progress toward agreeing

on or attaining a mutual goal. State 3 signifies that a promise or commitment has been made; state 5 signifies that the commitment has been fulfilled to the satisfaction of both conversants. Movement downward signifies either non-acceptance of a proffered agreement and a desire of negotiating, or refusal to engage further.

The conversation for action pattern is recursive, which is the quality that allows the goal-focus during an interaction to change from moment to moment. In the course of negotiating relative to a higher-level goal, a subordinate goal may have to be proposed, agreed, and satisfied for the higher-level goal to be satisfied. A breakdown can occur at any point, spawning a new subordinate goal that requires satisfactory attainment before progress can be claimed.

The conversation for action pattern provides a useful structure for instructional conversation and control design involving goal negotiation and satisfaction. As such, it supplies a pattern for control system design for instructional conversations. True to its philosophical basis, it respects the agency of the user, who may refuse offered goals and actions, make counter offers, and negotiate. In this respect, it is not an instructional theory specifying desirable degrees of freedom a user should be given: it is not a theory of learner versus instructor control. The conversation for action form can be viewed as a generic design theory defining how control design issues relate to the goals of human-computer or human-human interactions.

Gibson and Norman: Affordance Theory

Affordance theory was originally proposed by James Gibson, a researcher in the area of visual perception (Gibson, 2014). Gibson believed that the brain perceived the unchanging elements in its environment in terms of surfaces and edges that become recognized as objects by the reflection of light. Gibson proposed that invariant parts of a visual scene become objects that “afford” action or use. The parts of a scene that change provide contrast with those that remain stable, as the ever-moving eye scans the environment.

For Gibson, an affordance is the perceived object and “what it offers” (p. 56). Mankind arranges the environment “to change what it affords him” in terms of useful actions. Gibson suggests that “what we perceive when we look at objects are their affordances, not their qualities” (p. 58). He proposes that “what an object affords us is what we pay attention to” (p. 58).

Perceived objects that are other persons or animals not only can be acted upon, but they respond to action:

When touched they touch back, when struck they strike back; in short, they *interact* with the observer and with one another. Behavior affords behavior, and the whole subject matter of psychology and of the social sciences can be thought of as an elaboration of this basic fact.” (p. 58)

Donald Norman (1990, 1993, 2013) applied Gibson’s theory to explore the manner in which humans modify their environment to increase its utility and usability. He studied how human-made artifacts suggest (or not) how they might be used. His most familiar example of artifact affordances is the design of a glass door. Some doors, he noted, have identifiable elements that signal whether to push or pull, and where to push or pull, to open them.

Norman proposes that the ability of a user to operate within an affording (control) environment depends on the match between the designer’s conceptual model of the environment and the user’s conceptual model. He emphasizes, therefore, the importance of the match between the user and designer models.

According to Norman (1999), the appearance of a device could or should “provide the critical clues required for its proper operation” (n.p.). Norman emphasizes the importance of the “meaningfulness” of an affordance. An act is meaningful when a user acts based on shared cultural constraints and conventions: that is, on a language of control use shared within the culture of users.

Norman’s emphasis of the importance of the conceptual model possessed by the user defines a major difference between the psychological views of Gibson and those of Norman. Gibson denied the claim of cognitive psychology, that intermediary cognitive structures such as Norman’s “models” exist. This difference

of opinion was the source of a series of friendly discussions between the two psychologists over a period of time. The point of mentioning this is that the concept of affordance is useful to the designer, despite conflicts that might exist in its deeper explanatory-theoretical basis.

Norman (1993) names seven stages of taking an action:

1. Forming the goal
2. Forming the intention to act
3. Specifying the action
4. Executing the action
5. Perceiving the state of the world
6. Interpreting the state of the world
7. Evaluating the outcome

These stages should be considered by a control system designer, as they separate states of mind surrounding the actual use of controls. Only in stage 4 is a control actuated. Between stage 1 and stage 3 a goal becomes an intention and a means is selected, which includes a control to be used and, if applicable, the value of the control to be exercised or the content of the control assertion. Stages 5 through 7 include reading feedback from the action, determining whether the intended action was completed, and then determining whether the action was the right one and a satisficing one. This structure of stages in performing an action is very similar in form and intent to the TOTE (test, operate, test, exit) unit proposed in 1980 by Miller, Galanter, and Pribram, in *Plans and the Structure of Behavior* (1960).

Parrish and Laurel: Aesthetics of Control Systems

Control design is deeply involved in learning experiences that place learners within a role in a narrative space where they engage in the first-person or as avatar-actors. Controls in such settings are the “means” for “agents” to execute “actions” in the “setting” to help fulfill a “goal”: in this case, a goal aimed at learning (Parrish, 2007). This kind of application requires controls to become part of a larger aesthetic experience--navigating a space, exploring, solving problems, making decisions, and perhaps designing or creating an artifact in the process.

Designs of this kind string actions together with each other as part of the “telling” of the narrative. Therefore, “the object of an interface is a whole interaction, just as the object of a play is the whole action” (Laurel, 1986, p. 83). Individual actions have meaning only in relation to a specific point in the ever-changing narrative, rather than having discrete interpretations. While designing this type of learning experience, Parrish recommends that the designer write ‘design stories’ to determine the narrative of the experience from the user’s perspective (Parrish, 2006). This need not chart specific paths the user can follow, but it can involve that. Controls should allow learners to participate in an unfolding narrative and should lead to narrative unity (2007).

Controls in this environment are more closely related to the representational element of the interface because they blend with the fictional or imaginative aspect of the representation. They may be expected to function as an extension of the user’s role, which Laurel refers to as “first-personness”, which Laurel says is:

...affected by the representational aspects of the interface; that is, how the user’s choices and actions are introduced into the system, and how the activities of the system are represented to the user” (Laurel, 1986, p. 77).

In these cases, the experience itself may be part of the objective.

Laurel identifies three aspects important in the design of first-person controls: (1) interaction frequency, (2) interaction range, and (3) interaction significance (Laurel, p. 78). Frequency is a continuum representing how often user input is enabled. Range refers to the degree of constraint placed upon the controls offered.

Significance refers to the meaning or interpretation of control actuations at a given point in an ever-shifting context.

First-personness is most completely realized at the extreme end of each of the interactive variables' continuum: [where] frequency is continuous; range is infinite; [and] significance is maximal" (Laurel, 1986, p. 79).

Vincenti: Evolution of Control Theory Through Research and Development

Vincenti (1990) presents a detailed case study of the evolution of control design theory conducted over four decades. The case involves the discovery and solution of a control design problem that elucidated the process of theory creation. In the earliest days of the aviation industry, few researchers paid attention to flying qualities of aircraft.

As piloting experience accumulated during and after World War I, pilots began to refer to planes as having "light", "sluggish", or "firm" controls that made aircraft either easier or harder to fly, or in some cases dangerous. Comparisons between control systems accumulated as makes and models of aircraft proliferated. This made pilots aware of the variety of flying qualities and brought the problem of control design into focus. This led to "the notion that specifications could be usefully written for something as subjectively perceived as flying qualities.... It was not at all obvious nor obviously useful idea at the outset" (P. 52).

Vincenti's case study shows that a design community must detect control design as a problem, then identify the value of a solution to users and designers, then identify key variables and analytic concepts, and finally perform research to define boundaries of satisfactory control performance. The result of this process is a set of specifications for acceptable control systems.

Though the aviation control problem may not seem relevant to the design of computer control interfaces, one need only recall the struggle to define specifications for the Apple computer GUI (see Moggridge (2007)). As the sophistication of computer-delivered learning experience grows, there will necessarily be a similar process carried out to guide the work of educational interface designers. Part of the result will be a set of theories, norms, and standards for educational interface control design. For this reason, we suggest the aviation example is highly pertinent to learning experience controls.

In an earlier section we provided a composite view of control design principles gathered from the standards of many design fields. Virtually all of the qualities listed there are candidates as desirable qualities in learning system controls. Though there is not the same life-and-death consequence for educational controls that exists in some of the other fields, something even more important may be at stake: the costs to the underachiever that could be attributed to poor control design, viewed in their total impact across thousands of users, will be considerable. Just as the evaluation of operating system GUI standards has improved the efficiency of the computer user, reducing the time and attention required of the computer learner will increase learning productivity. If efficiency and depth of learning can be improved incrementally across a large population of educational computer users, the investment in the study of instructional control designs will be worth the effort it requires.

Vincenti identifies research stages through which a control problem is identified, eventually resulting in specifications. Instructional control system designers should take note of and anticipate the steps in this research process. In what follows we will try to apply lessons from the aviation community to the instructional design community.

1. Familiarization with the problem.

An initial step for the aviation community was learning "the true nature of the airplane" (p. 102). Though airfoils and control surfaces (and their associated cockpit controls) could be designed, they could not always

be made safely flyable, as the Wright Brothers learned. Unexpected combinations of forces could de-stabilize an otherwise stable aircraft, resulting in sudden loss of control. Likewise, we feel that "the true nature of the learning experience" has yet to emerge.

2. Identification of basic variables and derivation of analytical concepts and criteria.

Aviation had to discover basic variables of control systems by flying and by--over a period of time--amassing mountains of data on aircraft performance for different aircraft designs. Only after much flying data had been related to flying quality reports could quality concepts (descriptions, subjective judgments) emerge into the literature, as measurable variables and criteria.

According to Vincenti:

Identification of criteria included such things as stick force and elevator angle as pertinent variables, and long- and short-period modes as concepts in dynamic stability, and stick force per g as a criterion of control. The issue here was how to represent flying qualities quantitatively in engineering terms (p. 102).

It should be noted that as technical and impersonal as these terms make it sound, the emphasis in control system research was on creating aircraft systems that were safer, measurably easier to use, and subjectively more agreeable to the user. In this simplified sense, study of instructional controls and their application should be expected to have a similar impact.

3. Development of instruments and techniques for measurement.

Instrumentation, the emergence of analytical concepts, and measurable variables emerged in the aviation industry in a cycle of mutual interaction. Measures were centered around dynamic qualities, such as aircraft stability moment-to-moment. The challenges of aircraft and instructional control-related measurement are similar in that they deal with control within a dynamic, changing environment. They are both problems of measuring and maintaining stability within an environment, keeping in balance a tenuous stability that is always only seconds away from vanishing.

The issue of control, therefore, is not just a matter of accepting and processing individual control inputs, but rather one involving feedforward and feedback cybernetic processes of continuous tracking, adjusting, correcting, and anticipating of a trajectory. The new research direction in learning analytics can give an edge to instructional design studies over the early aeronautical studies of instrumentation because of the computer's increased ability to capture enormous amounts of data and sift them for patterns using sophisticated statistical techniques. However, the results of intensive learning analytic studies will only be as useful as the categories anticipated by the variables, the analytic concepts, and the granularity of the data captured. It is unlikely that the current appreciation of "the true nature of the instructional experience" will supply an adequate basis for real progress. It is likely that new concepts of dynamically balanced processes of learning and instruction will be required, just as they were required in aviation.

4. Growth and refinement of user opinion regarding desirable operating properties

In aviation the number of pilots grew rapidly in both wartime and post-wartime environments. Similarly, we can expect the number of computer-using learners to surge in the near future. Pilots were not aeronautical engineers, but their experience and their response to aircraft and control design, nonetheless provided important clues for bridging the worlds of the engineer and the user, just as will the judgments of the learner:

The reasons why an airplane must be stable or more or less obvious. A moderately unstable airplane may be flown safely if the pilot continually manipulates the controls. However, this procedure is very nerve-wracking and tiring. Landing or take-off is dangerous in an unstable ship. In flying blind, it is a great comfort to know that if the controls are released, the ship will continue on its course safely even in rough air (p. 72-3).

Vincenti notes that in 1910 this had been a controversial idea, but that by 1935 it had become "obvious". Growth in the number of computer-based learning users may similarly turn controversial ideas regarding instructional control system design into "obvious" principles to guide design. This will probably happen in a response to user opinions that are shared by users through social media "likes" and product reviews. Control design principles will probably be shaped as much by user preferences as by psychological or theoretical principles, and from patterns of user preference there will most likely emerge new theories for control design.

This should not be taken as an excuse to avoid theorizing about instructional system control design under dynamic learning conditions, because over the years the convergence of the engineer's world and the user's world grew together as demands for increasingly improved performance escalated. Vincenti describes how disagreements between pilots and engineers were resolved in one case:

[A group of pilots and engineers] turned themselves into research pilots and flight-research engineers. The process was probably an unconscious one that took place naturally out of the demands of the job (p. 76).

In another instance a similar pattern was noted in the cooperation of two famous test pilots, one of whom was by training an engineer and one of whom was by experience a test pilot:

Mel Gough...was very much interested in handling qualities. He didn't know too much about why an airplane did what it did, but he knew what he liked. He used to take me up and show me what airplanes would do. When he got an airplane that was rogue, he'd say "I want you to come up because I want to show you this. Maybe you can tell me how to fix it." And so we did a lot of flying together. In fact we even flew one time in a single seater. I had a foot on one pedal and he had a foot on the other (Interview with Gilruth, 1986, quoted by Vincenti, p. 91).

The increasing importance of user opinion in aviation control design may have an analog in instructional control design. Factors cited by pilots included fatigue experienced during longer flights, during flights where continuous manipulation of controls was required to maintain flight regime, higher performance systems (e.g., speed), increased blind flying due to conditions (e.g., night, weather), increasing peripheral duties (e.g., communications). Vincenti explains, "the art of flying had grown up and become more complex" (p. 73).

Likewise, should we not anticipate a similar pattern with instructional trends: longer sessions of instructional interaction (fatigue factor), increased rate or intensity of responding (response frequency factor), more sophisticated products (performance factor), increased variety of instructional places and conditions (environmental factor), and demands for increased multitasking (peripheral duties factor)? As these factors vie for the learner's attention and energy, instructional control systems must become increasingly transparent in ways that may not be readily apparent to either the designer or the user but that will require designer-users-in-one-skin to bridge the gap. As things stand today, surprisingly few designers of computer-based learning systems have themselves been extensive users of the kinds of products they design for others. In aviation:

How the various influences became translated into books and articles is not the kind of thing that appears in the record. It must have involved pilots, designers, research engineers, and academics talking to each other and participating to some extent in each other's activities. However, it took place, it reflected a complex and widespread learning process by considerable engineering community (p. 73).

5. A deliberate scheme of research.

After decades of uncertainty the aviation industry converged on a focused research agenda, prompted by a joint cooperative effort of manufacturers and government and industry organizations. The multi-year

research plan was seeded by an aircraft specification that for the first time included a detailed section on required flying qualities. The specification "embodied for the first time the notion that desired subjective perceptions of pilots could be attained through objective specifications for designers" (p. 81). The specifications appear to have been arrived at after consultation with airline pilots, engineers, and researchers.

In a sense, this marked the transition from what had been mainly academic research, to a program of cooperative R&D that brought together engineers and researchers representing manufacturers, the academy, government, professional organizations, and users. A similar effort to focus educational design is unprecedented, except for the period following the launch of Sputnik in 1957. The focus behind the aviation research program was sharpened by the clear definition of the research process, necessitated by inter-organizational coordination. The research plan followed exactly the stages outlined by items #2, #3, and #4 above. The research program was further focused because it was carried out with respect to a design of a specific aircraft.

As aircraft development and programmatic research proceeded in parallel, research results fed back into the specification to improve it. Vincenti notes that this is typical of current engineering research:

...Engineering research typically takes place within a context of practical demands. Since the demands usually will not wait, research engineers are often called upon to give advice when their knowledge is still at a formative stage. Doing so requires them to articulate and sometimes alter and improve their developing ideas. Such constructive exchange between the generators and users of engineering knowledge is an essential element in the learning process (p. 89)

In the end, several things were accomplished by this research program, the most important of which relates the present theme: that control aspects of the technological design have great importance, and that what might be considered subjective qualities of the control system design can be subject to empirical research, development, and the application of theory, just as can their physical and mechanical properties and functions. The main problem for aviation control designs beyond this point became "mainly one of designing (i.e., proportioning) the airplane rather than deciding at the same time what to design *for*" (p. 97, emphasis in the original).

Conclusion

This survey of control design theory and a theory-building case study illustrates a wide range of control-related issues from the designer's point of view. It shows that there exists a substantial body of relevant theory that can improve educational experience designs at a fundamental level. Designers tend to take control design issues for granted, and researchers often neglect them. We believe the issues of the control layer deserve to be established in the instructional design literature, if for no other reason than that significant instructional interactivity depends on appropriate control designs. Controls are the means by which learners practice the use of nascent knowledge. Controls are what enable the learner to act in significant and relevant ways as learning takes place. Control design should be one of the first considerations of a design, not an afterthought.

The umbrella term "interface" has centered our thoughts on the visible, surface features of interface designs, rather than on the exercise of knowledge provided by controls. The existence of control-related theory suggests that that we should focus more attention on how to engage the learner in significant mental processing through intelligently designed controls. This would constitute a first step in unpacking the interface concept. We suggest that further steps should include considering layers of message and representation separately as well, to discover the theoretical and practical questions these raise for the designer and what can be learned from other design fields about these layers of a design.

As designs in other fields have become more sophisticated, control design theories have become important for designing control systems with properties matched to the learning task. Maturing intellectual concepts of instructional design and rising consumer expectations of instructional experiences will in the future make it

necessary for instructional designers to become more familiar with the theoretical bases of their control system designs.

References

- American Society for Testing and Materials (ASTM), (2013). *Medical devices and medical systems--Essential safety requirements for equipment comprising the patient-centric integrated clinical environment (ICI)--Part I: General requirements and conceptual model* (ASTM F2761-09 (2013)). Available from: <http://www.astm.org/Standards/F2761.htm> .
- Anderson, R. C. (1967). Educational psychology. *Annual review of psychology*, 18, 129-164.
- Austin, J. L. (1962). *How to do things with words*. Cambridge, MA: Harvard University Press.
- Baldwin, C. & Clark, K. (2000). *Design Rules: The power of modularity*. Cambridge, MA: The MIT Press.
- Benjamin, L. T. (1988). A history of teaching machines. *American psychologist*, 43(9), 703-712.
- Brand, S. (1994). *How buildings learn: What happens after they're built*. New York: Penguin.
- Bruner, J. (1966). *Toward a theory of instruction*. Cambridge, MA: The Belknap Press of Harvard University Press.
- Carlisle, R. P. (1997). The relationship of science and technology: A bibliographic guide. A joint publication of the Navy Laboratory/Center Coordinating Group, and Department of the Navy (ISBN 0-945274-38-6).
- Collins, A., Brown, J. S. & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (Ed.) *Knowing, learning and instruction: Essays in honor of Robert Glaser*. Hillsdale, NJ: Erlbaum.
- Constant II, E. W. (1984). Communities and hierarchies: Structure in the practice of science and technology. In R. Laudan (Ed.), *The nature of technological knowledge: Are models of scientific change relevant?*. Dordrecht, GR: D. Reidel Publishing Company (Kluwer), 27-46.
- Crawford, C. (2003). *The art of interactive design*. San Francisco, CA: No Starch Press.
- Dewey, J. (1929). *The sources of a science of education*. New York: Horace Liveright.
- Dickmeyer, N. (1989). Metaphor, model, and theory in education research. *Teachers college record*, 91(2), 151-159.
- Dorst, K. (2010). The nature of design thinking. In K. Dorst, S. Stewart, I. Standinger, B. Paton and A. Doug (Eds.), *Proceedings of the 8th DTR Symposium: Interpreting design thinking*. Sydney, AU, 19-20 October 2010 (ISBN 928-0-9808622-2-5).
- Dorst, K. (2015). *Frame innovation: Create new thinking by design*. Cambridge, MA: The MIT Press.
- Duffy, T.M. & Cunningham, D. J. (1996). Constructivism: Implications for the design and delivery of instruction. In D. Jonassen (Ed.), *Handbook of research for educational communications and technology (1st ed.)*. New York: Simon & Schuster Macmillan, 170-198.
- Ertmer, P. & Newby, T. (1993). Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. *Performance improvement quarterly*, 6(4), 50-72.
- Fowler, M. (1999). *Refactoring: Improving the design of existing code*. Boston, MA: Addison-Wesley.
- 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, Part I*, 268-285.

Gibbons, A. S. (1997). The TREKKER: Unifying metaphor for interface design. Presented at the Instructional Technology 9th Annual Summer Institute, Instructional Technology Department, Utah State University, Logan, UT.

Gibbons, A. S. (2014). *An architectural approach to instructional design*. New York: Routledge.

Gibbons, A. S. & Bunderson, C. V (2005). Explore, explain, design. In K. Kempf-Leonard (Ed.), *Encyclopedia of Social Measurement*. New York: Elsevier, 927-938.

Gibson, J. J. (2014/1979) The Theory of Affordances. In J. J. Giesecking, W. Mangold, C. Katz, S. Low, & S. Saegert, *The people, place, and space reader (1st ed.)*. New York: Routledge.

Green, P. Levison, W. Paelke, G. & Serafin, C. (1994). *Suggested human factors design guidelines for driver information systems*. Technical Report FHWA-RD-94-087, Office of Safety and Traffic Operations R&D, Federal Highway Administration, McLean, VA.

International Organization for Standards (2012). *ISO 4040:2009: Road vehicles--Location of hand controls, indicators, and tell-tales in motor vehicles*. Available at: www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44856 .

Josephson, J. R. & Josephson, S. G (1996). *Abductive inference: Computation, philosophy, technology*. Cambridge, UK: Cambridge University Press.

Klir, G. J. (1969). *An approach to general systems theory*. New York: Van Nostrand Reinhold.

Laurel, B. (1986). Interface as mimesis. *User centered system design: New perspectives on human-computer interaction*, 67-85.

Lave, J. & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge, UK: Cambridge University Press.

LEGO Mindstorms (n.d.). Available at: www.lego.com/en-us/mindstorms/?domaindir=mindstorms.lego.com .

Lehman, K. (2011). Profile of Stephen Hawking. *Science and technology libraries*, 30:1-19.

Lowyck, J. (2014). Bridging learning theories and technology-enhanced materials: A critical appraisal of its history. M. Spector, M. D. Merrill, J. Elen, and M. J. Bishop (Eds.), *Handbook of research on educational communications and technology (4th ed.)*. Berlin: Springer.

Maloney, J. Resnick, M., Rusk, N., Silverman, B. & Eastmond, E. (2010). The Scratch programming language and environment. *ACM transactions on computing education*, 10(4).

Mandler, G. (2002). Origins of the cognitive (r)evolution. *Journal of history of the behavioral sciences*, 38(4), 339-353.

Markle, S. M. (1967). Empirical testing of programs. In P. Lange (Ed.), *Programmed instruction: The sixty-sixth yearbook of the National Society for the Study of Education, Part II*. Chicago, IL: University of Chicago Press.

Martin, R. L. (2009). *The design of business: Why design thinking is the next competitive advantage*. Cambridge, MA: Harvard Business Review Press.

McNerney, T. S. (2004). From turtles to tangible programming bricks: Explorations in physical language design. *Personal ubiquitous computing*, 8, 326-337.

Medeiros, J. (2015). How Intel gave Stephen Hawking a Voice. Available online at: <http://www.wired.com/2015/01/intel-gave-stephen-hawking-voice/>).

- Mehta, A. (2007). When a button is all that connects you to the world. In A. Oram and G. Wilson (Eds.), *Beautiful code: Leading programmers explain how they think*. Sebastopol, CA: O'Reilly Media.
- Miller, G. A., Galanter, E. & Pribram, C. (1960). *Plans and the structure of behavior*. New York: Holt.
- Moggridge, B. (2007). *Designing interactions*. Cambridge, MA: The MIT Press.
- Norman, D. A. (1990). *The design of everyday things*. New York: Doubleday/Currency.
- Norman, D. A. (1993). *The design of everyday things*. Cambridge, MA: The MIT Press.
- Norman, D. A. (1999). Affordance, conventions, and design. *Interactions*, 6(3), 38-43.
- Norman, D. A. (2013). *The design of everyday things: Revised and expanded edition*. New York: Basic Books.
- Oviatt, S. (2013). *The design of future educational interfaces*. New York: Routledge.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.
- Parrish, P. (2006). Design as storytelling. *Tech Trends*, 50(4), 72-82.
- Parrish, P. (2007). Aesthetic principles for instructional design. *Education technology research and development*, 57(4), 511-528.
- Resnick, M. (1997). *Turtles, termites, and traffic jams: Explorations in massively parallel microworlds*. Cambridge, MA: The MIT Press.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the Association for Computing Machinery*, 52(11), 60-67.
- Resnick, M., Ocko, S. & Papert, S. (1988). LEGO, LOGO, and design. *Children's environment quarterly*, 5(4), 14-18.
- Scardamalia, M. (2004). CSILE/Knowledge Forum(r). In *Educational Technology: An encyclopedia*. Santa Barbara, CA: ABC-CLIO, 183-192. (Available from: http://learnteachlead.ca/wp-content/uploads/2014/11/CSILE_KF-2.pdf)
- Scardamalia M. & Bereiter, C. (1994). Computer support for knowledge-building communities. *Journal of the learning sciences*, 3(3), 265-283.
- Schon, D. (1987). *Educating the reflective practitioner: Toward a new design for teaching and learning the professions*. San Francisco, CA: Jossey-Bass Publishers.
- Simon, H. (1999). *Sciences of the artificial (3rd ed.)*. Cambridge MA: The MIT Press.
- Venable, J. R. (2006). The role of theory and theorising in design science research. Presented at the first International Conference on Design Science, Research in Information Systems, and Technology. Claremont, CA: Claremont Graduate University.
- Vincenti, W. G. (1990). *What engineers know and how they know it*. Baltimore, MD: Johns Hopkins University Press.
- Walton, D. (2005). *Abductive reasoning*. Tuscaloosa, ALA: University of Alabama Press.
- Wertsch, J. (1998). *Mind as action*. Oxford, UK: Oxford University Press.
- Wilson, B. G, Jonassen, D. H. & Cole, P. (1993). Cognitive approaches to instructional design. In G. M. Piskurich (Ed.), *The ASTD handbook of instructional technology*. New York: McGraw-Hill, 21.1-21.22.

Winograd, T. & Flores, F. (1987). *Understanding computers and cognition: A new foundation for design*. Reading, MA: Addison-Wesley.

Yeh, M., Jo, Y., Donovan, C., & Gabree, S. (2013). *Human factors considerations in the design and evaluation of flight deck displays and controls (DOT/FAA/TC-13/44; DOT-VNTSC-FAA-13-09)*. Washington, D. C.: Federal Aviation Administration.