Information Architecture Without Internal Theory: An Inductive Design Process

Marsha Haverty
AGENCY.COM, Ltd., 1805 SE 44th Ave., Portland, OR 97215. E-mail: mhaverty@acm.org

This article suggests that Information Architecture (IA) design is primarily an inductive process. Although top-level goals, user attributes and available content are periodically considered, the process involves bottom-up design activities. IA is inductive partly because it lacks internal theory, and partly because it is an activity that supports emergent phenomena (user experiences) from basic design components. The nature of IA design is well described by Constructive Induction (CI), a design process that involves locating the best representational framework for the design problem, identifying a solution within that framework and translating it back to the design problem at hand. The future of IA, if it remains inductive or develops a body of theory (or both), is considered.

Introduction

Information Architecture (IA) as a process involves creating solutions to design problems. Recent discussion in the IA community suggests that Information Architecture is in part a bottom-up process. The bottom-up part is usually associated with creating metatags for content in dynamic databases and technologies like XML (Bullard, 1999; Merholz, 2001; Rosenfeld, 1998). But content components involve just one aspect of the design problem—navigation and other interactive components must also be addressed. Navigation is designed specifically for regions or subregions, then tied together through contextual links; interactive tools are created as self-contained modules, then integrated into the context of the surrounding architecture. Once all content, navigation, and interactive components are designed, they become building blocks working up to the overall Information Architecture of a system. Considering this collection of bottom-up design activities, the design process of Information Architecture may be considered an inductive process.

The Information Architect periodically shifts to a top-down perspective to ensure that each building block meets high-level goals, user attributes, and available content. But, the moments of design usually happen on a small scale, and address one very specific design problem at a time.

Why Information Architecture Is Inductive

The easy explanation for why Information Architecture is an inductive process is simply because it is a new field that lacks its own body of theory. A formal discipline (e.g., biology) is a branch of knowledge and teaching grounded in a deep theoretic basis; theory guides the activities of workers in the community. A field is an area of academic interest or specialization that draws on theory, but usually from other related disciplines. Lack of internal theory leaves workers in a field to start from scratch with each design problem. As a field matures and workers gain experience, useful solutions may be reused, but confidence in the ability of the solution to fit the given context and fulfill the problem is still tentative and perhaps even unreliable without formal validation. Various methods (e.g., usability testing) can help verify that a solution is well suited to a particular situation, but generalization to other contexts is a case-by-case process without theory. What theory can do is shrink the space of possible solutions to those known to be appropriate for the design problem—what is left to the worker is to contextualize the theory to fit the specifics of the design problem.

Information Architecture may be considered a field, but has not yet reached the status of discipline. Currently, the IA community supports a variety of definitions for what it does and long lists of related disciplines that inform its work (ASIS Summit, 2000, 2001; SIGIA-L, 2001). In fact, so many related disciplines and fields have the potential to inform IA that the field is often considered unbounded (V-2 Organization, 2001). This implies that Information Architects are well served to feed their curiosity and explore seemingly related fields in search of design solutions, tracking and reusing what works along the way. Fortunately, this openness is what brings many people to the field of IA in the first place.

The hard explanation for why IA is an inductive process is because it supports emergent phenomena. A system ex-
hibits emergence when a small set of building blocks, constrained by simple rules, can generate a huge number of complex patterns. An emergent phenomenon cannot be summarized by a description of its individual parts; the whole is not equal to the sum of the parts (Holland, 1998). For example, in the game chess, the building blocks are chess pieces (pawn, rook, king, queen, etc.), and there is a simple set of rules for how the pieces may move on the board. But an understanding of the interactions of the pieces that emerges during play is what is needed to play chess. As Holland (1998, p. 14) explains:

We cannot get a representative picture of a game in progress by simply adding the values of the pieces on the board. The pieces interact to support one another and to control various parts of the board. This interlocking power structure, when well conceived, can easily overwhelm an opponent structure with higher-valued pieces that are poorly arrayed.

In fact, only after time and intense study can patterns begin to be identified (expert chess players refer to high-level patterns such as “fork” and “pin”). Chess is defined by less than 24 rules, but gives rise to uncountable patterns and possibilities.

Even the first hypertext visionary, Vannevar Bush, had a sense of emergence. Bush describes a synergy that comes from the combination of human action and machine automation. He suggests that the essence of this interaction cannot be explained solely through an understanding of each part (Bush, 1945).

Considering Information Architecture as the basis for user experience, it is another case of building blocks giving rise to emergent phenomena. Information Architecture consists of a finite number of content, navigation, and interactive components. However, the real nature of the IA does not present itself until users interact with the system. It would be impossible for an Information Architect to predict the entire set of possible user interactions—each user has a different experience, and the same user will have different experiences with each session. Although user experience is something greater than the IA, it is directly dependent on it. Information Architects define the basic framework that constrains the set of possible interactions. The framework is not a prescription for user interaction, but a constraint.

Usability testing and other methods can begin to study the experience that emerges with user interaction, but IAs cannot design navigation, content, and other interactive components knowing exactly how each one will contribute to the overall experience. This is not to say that it is impossible to develop theory to address emergence at a higher level—the level of regularities in patterns of interaction—but methods for how to study and control emergent phenomena are just now being developed (Holland, 1998).

To summarize, the design process of Information Architecture is inductive for two reasons: (1) IA does not have internal theory to guide top-down design and, therefore, must proceed by bottom-up, or inductive design; (2) Information Architectures support the emergent phenomena of user experiences, and until we better understand how to study and design for emergent phenomena, they will require bottom-up designs.

At this point, we see why IA design is an inductive process, but the next question is how does the process work?

Nature of the Process: Constructive Induction

It has just been suggested that, beyond making modular content, the entire process of IA design is inductive (periodically stepping back to validate design work against top-level goals, user attributes and context). Borrowed from machine learning and engineering, Constructive Induction (CI) seems to describe the inductive process of IA fairly well. Constructive Induction is a process for generating a design solution using two intertwined searches. The first search involves identifying the most adequate representation framework for the problem; the second search involves locating the best design solution within the framework and translating it to the problem at hand (Arciszewski, Michalski, & Wnek, 1995; Bloedorn & Michalski, 1998).

Constructive Induction was originally developed for machine learning algorithms, but has also been applied to engineering design and design creativity (Arciszewski et al., 1995). In the context of design problems, CI is useful when the representation space is inadequate for the problem (as is the case for new fields without their own body of theory), or when existing theory is not robust enough to handle new design problems.

Because IA does not have its own body of theory, Information Architects borrow from other fields and disciplines, and that borrowing is unbounded. So, just like in CI, a major activity in IA is locating the best framework for the design problem, identifying the best solution within that framework and repurposing it to fit the context. These steps appear in Figure 1.

Step 1

In the first step of the process, the IA determines the overall goals for the system (such as the client’s vision for the system, business or educational requirements, etc.), user attributes (user goals and characteristics of target users of the system), and available content for the site (such as copy and images). The IA must then define the design problem and break it into basic design problems, or pieces of the problem that require specific design solutions. The IA may use tools like sitemaps (high-level diagrams showing how the main sections of the system are hierarchically arranged) to help track the overall plan for the system as specific pieces are addressed. But, the actual design takes place in the form of finding solutions for each basic design problem.

For example, suppose an IA is designing a system for students to study and research a large body of complex information. The first thing the IA does is determine the goals of the teacher or industry funding the system, the
goals of the students as users of the system, and the available content for the system (textbooks, research papers, images, etc.). With that background, the IA articulates the basic design problems for the proposed system. Suppose one of those design problems is the need for students to be able to research and answer a question with a large body of complex information, and, in future sessions, they also need to study the paths they took to come to their answers.

Once all of the basic design problems are articulated, the IA moves to the next step of the design process.

Step 2

The next step involves searching for the best representational framework to address each basic design problem (the first search outlined in the Constructive Induction process) and identifying a solution within that framework (the second search in the Constructive Induction process). The IA considers the nature of the design problem and identifies an appropriate representational framework that might yield a solution.

In the example of an academic system for students (described in Step 1), one basic design problem is that students need a way to return to and study the paths they originally took to answer questions in future sessions. The IA might decide that this problem seems to fall within the fields of wayfinding and educational technology. The IA would look to the research in these fields for a design solution.

Selecting appropriate frameworks and solutions is much more effective with immersion in the design problem and familiarity with other related disciplines: “Having a familiarity with several ‘nearby’ disciplines, when the target does not fit well within an established discipline [or when there is no discipline], will enhance the possibility of a source/target transfer” (Holland, 1998, p. 213). Although solutions are often decontextualized and transferred to a novel context (Davies, 1989), they do not arise randomly—they must be driven: “Many creative thinkers testify that they have pre-
vously immersed themselves deeply in the subject matter of the problem, often over considerable periods,” (Gagne, 1977, p. 164). This immersion is often accompanied by a tension, or a need to know that drives the process, and can also be motivated by an aesthetic sense (Ladd, 1987).

As has been described for Information Science (Bates, 1999) and for professional activities in general (Polanyi, 1966), workers in a community have some level of tacit knowledge that goes unstated by the community. Although IAs cannot predict the experience that will emerge with user interaction of a system, tacit knowledge and insight come in the form of actively selecting quality design solutions from related disciplines and applying them in ways that elegantly address the design problem.

In some cases more than one framework may inform a solution, and one solution may address more than one basic design problem. Also, not every design problem will require this detailed search—sometimes solutions will be used based on experience, best guesses, or external requirements.

Once the design solution has been identified, the next step is translating it back to the design problem.

**Step 3**

The activity of translating a solution from a representation framework back to the design problem (closure for the second search in the CI process) happens directly following, or sometimes in conjunction with the activity of identifying the applicable solution (described in Step 2). (A similar translation activity is also characterized for Information Science (Bates, 1999).) Steps 2 and 3 are the interrelated creative activities that are not necessarily linear or algorithmic, but must be described here as a “steps” for coherent discussion.

In the academic system design example, suppose the IA locates the concept of “landmark” in the way-finding literature and decides it could apply to the problem of allowing students to revisit and track exact pathways through academic information. The next step is actually translating the concept of “landmark” into an Information Architecture component and integrating the component into the system.

**Step 4**

Once solutions have been translated to the current design context for all basic design problems, the IA must integrate them to form the overall Information Architecture. Throughout this process, the design solutions are occasionally validated against original goals, users, and available content. Activities like team reviews and usability testing may help achieve this validation.

**Avoiding Reductionism**

The idea of breaking complex systems into components hints at reductionism. When you reduce things to their building blocks, you often lose general properties that emerge when they are considered together as a whole (remember the chess example where the game cannot be understood by listing the available pieces and rules for how they can move, but by seeing the interaction of all the pieces on the board as they change throughout play). Focusing on the quality of the building blocks is the right approach for building systems that exhibit emergence, but focusing on building blocks would be an incomplete approach for assessing the quality of the system once it is built.

Although IAs worry about building blocks when they create solutions to design problems, they focus on user experience when they evaluate the finished design. The overall IA weaves across all of the individual design solutions to provide the framework for user interaction. The quality of an Information Architecture is discussed in terms like *ease of use* and *satisfaction*. These are holistic, top-level requirements of the system that may be instantiated by many possible combinations of building blocks. The important thing is that the design solutions, taken together, meet these high-level criteria. Also, design solutions may be more or less elegant, depending on immersion and familiarity with nearby disciplines, and the solutions are periodically measured for consistency with overall goals, user attributes and content, better assuring their quality.

Even if the IA design process changes greatly in the future, IAs will probably still predict how well a design supports user experience at the level of patterns of interaction, not by reducing the system to individual elements.

**Future: Theory or Induction (or Both)?**

Picture the following three scenarios for the future of IA.

**Scenario 1: IA Remains Inductive**

If Information Architecture does not develop theory and remains a primarily inductive process, it will continue to be a challenge to establish metrics for assessing the quality of IA systems. The field may continue to develop best practices or design patterns for common design problems across many contexts. Much work has already been done in this direction (ACM-SIGWEB, 2001; Huncher, 2000; Tidwell, 1999). The more that is learned about specific patterns, the more likely it is that higher level patterns spanning several levels of abstraction will be recognized and applied to more robust design problems. In addition to usability testing, IAs may develop other ways to understand and plan for the emergent systems they create.

Over time, design problems may commonly draw solutions from the same pool of nearby disciplines, but certain design problems may lend themselves to disciplines that have not yet been explored. Without a guiding theory, the possibilities for innovation and creativity are unbounded. This openness and room for innovation is what may continue to attract many IAs to the field.
**Scenario 2: IA Develops Theory**

If theory specific to IA was developed, how would it affect the design process? With guidance from theory, IAs could begin with a top-down approach and fill in the specifics to contextualize the theory to the situation. But, there may still be times when the IA must work inductively to fill gaps not addressed by theory. The overall top-down approach of Information Architecture guided by theory is discussed in Figure 2.

**Step 1**

As usual, the Information Architect would consider the overall goals, user attributes, and available content, and determine the basic design problems.

**Step 2**

Instead of engaging the two-step inductive process of locating an appropriate representational framework then identifying a solution and translating it back to the current context, the IA would look to theory. IA theory would provide parameters by which appropriate design solutions may be identified. Theory might provide for holistic solutions that address more than one basic design problem. Solutions would be generalized, so the IA would still require immersion in the design problem and tacit insight to contextualize the theory for the given situation in elegant ways.

**Step 3**

The IA would still weave design solutions together to form the overall Information Architecture. Overall, IA design would be guided by a top-down theoretical perspective and become a primarily deductive process. The finished system would still need to be tested for usability, but the solutions would already have some theoretical basis for their goodness of fit to the problem.

IA theory would be helpful to the design process, but what would it look like? Does “sameness” even exist in IA
outside of specific contexts? Maybe the body of knowledge that makes IA a discipline would be a collection of design patterns that could be reused, depending on the situation. A pattern for IA if it remains inductive (described in Scenario 1) would be a design solution that has been identified by the IA community as a useful approach to a specific design problem, has stood the test of time, but may or may not apply to each new context. Alternatively, a pattern that contributes to the theoretical basis for IA as a discipline would be a design solution that has been identified by general research, has stood the test of time, and is highly likely to apply to new contexts.

Because there is a variety of solutions that may be created for any given design problem, an IA discipline may allow for differing schools of thought, or local theories. The concept of macrotheory has been suggested to avoid this division in the field of Human–Computer Interaction (HCI) (Barnard, 2000). Macrotheory is a new kind of multilevel theory. For HCI, it is suggested as an overlapping, layered structure of macro- and microtheories based around generic “systems of interactors” (Barnard, 2000).

Even if the complex design process of IA could be guided by theory, there is the issue of change. Within a discipline with a deep body of theory, there are long periods of “normal science” in which one paradigm provides the overriding theory that the community takes as a given to inform all work (Kuhn, 1996). However, a paradigm shift occurs when the theory is no longer robust, or when irregularities or eccentricities become more frequent and cannot be addressed by existing theoretical explanations without substantial edits. At this point, many small revisions to existing theory are not enough—a new theoretical paradigm is required.

If IA becomes a discipline, it will also be subject to periods of “normal science” and paradigm shifts. However, when we think about how fast IA is changing right now (especially considering developments in new technologies like XML, new hardware interfaces like wireless devices, and new interaction modes like voice interaction), change may be so rapid that the mode of agreement about which solutions work may need to be consensus within the community rather than the establishment of theory. Communities are actually very good at encompassing complexity, multiple view points, and change that go along with growing, unbounded, and in some senses unstable fields (Romero, 1998).

The feasibility of theory development is left unknown until the nature of IA theory is understood.

Scenario 3: All of the Above

Maybe IA could support some combination of general theory and an inductive design process that supports complexity, innovation, creativity, and rapid change. If IA develops some initial theory, the theory could be added to, revised, and reworked with each new system through theory-guided constructive induction. Theory-guided constructive induction is a process in which the initial theory is analyzed and new features are constructed based on the components of the theory (Donoho & Rendell, 1995). This might allow for the rapid change that exists in IA now and possibly in the future.

There is also the possibility that IAs may be able to study patterns not in terms of general research or agreement on best practices, but by means of complexity theory. Complexity theory is just now offering ways to plan for, model, and study emergent phenomena. The ability to recognize the essential features and regularities of patterns abstracted out from various instances is the key to understanding and modeling emergence (Holland 1998). In the case of IA, work may also be informed by hypertext theory that focuses on the abstract topological structure of hypertext independent of content and context [e.g., Bernstein’s reusable hypertext patterns (1998), or Kolb’s regions and contours of self-representation (1997)], and may provide useful interaction patterns.

It is an open question as to which of these possible futures for IA is more likely than the others, but by acknowledging the possibilities, the IA community may actively work toward a desirable scenario. The development of academic programs and research in IA depend on this.

Conclusions

Information Architecture design is inductive because IA is a field without theory, and because it involves the design of building blocks that support the emergent phenomena of user experiences. The IA design process resembles Constructive Induction, a two-step inductive process where the best representational framework for the problem is located, then a good solution within that framework is identified and translated back to the design problem. IAs periodically step back and evaluate how building blocks might add to the overall goals for the system, but design activities mainly live at the level of researching and creating the building blocks and integrating them into an overall Information Architecture. This may change in the future if IA develops a body of theory, but accounting for rapid change would be an issue.

The thing that draws some Information Architects to their field is the very thing that offers such a struggle in terms of shaping it as a discipline: the open, unbounded nature of the emergent phenomenon of user experience. Whether IA becomes a discipline or continues to operate as a field, the idea of how to design for and study emergent phenomena will be a key topic.

References


round-table conference on computational models of creative design, December 3–7, Heron Island, Queensland, Australia.


