COMPLEXITY, CYBERNETICS, AND INFORMING SCIENCE: BUILDING A BETTER MOUSETRAP[1]

T. Grandon Gill
Information Systems & Decision Sciences Department, University of South Florida
Tampa, FL 33620 USA

ABSTRACT

Our decision-making and task environments are driven by three forms of complexity: complexity as we experience it internally (e.g., difficulty, uncertainty, ambiguity), complexity as it relates to our symbolic representation of tasks and plans (e.g., number of paths, program size), and complexity as a description of the decision environment and its behavior (e.g., ruggedness, turbulence). When experiencing high levels of complexity, we respond by constructing informing systems that better connect us together and offer increasingly rapid access to more information sources. In doing so, however, we inadvertently feed a cybernetic loop that leads to ever-expanding complexity (in all three forms). Left unchecked, this loop has the potential to alter both the way we think and the environments we face in ways that we may not desire.

Building a better mousetrap requires us to rethink both our approach to education and to designing systems. On the education side, we need to spend less time emphasizing specific content and more on building the student's the ability to react to complexity in ways that do not rely on making the world more complicated. On the design side, systems must increasingly emphasize adaptability as opposed to efficiency.

Keywords: Task complexity, informing science, goal setting, cybernetics, imitation, artifacts, problem space.

1. INTRODUCTION

The term "complexity" means different things to different people. To some, it refers to a mental state evoked by a context; a close relative to difficulty, uncertainty and ambiguity. To others, it describes the characteristics of a strategy or program: the number of possible paths, the amount of knowledge required, its potential for error. Still others view it as an objective feature of the task environment, using expressions such as turbulence or ruggedness. What these three perspectives share in common is one key element: the belief that becoming better informed is likely to be the ultimate antidote to complexity. Towards this end we construct informing systems that serve to provide decision-makers with faster access to higher quality information and offer direction for the choices that must be made. As these systems are utilized, however, we often fail to consider the impact that they exert on the decision environment itself. The result can be a cybernetic loop that leads to ever-increasing complexity of all forms.

To understand this phenomenon, we must first examine the nature of complexity—particularly task complexity—and the three broad ways in which it is most commonly defined. The interaction between these three domains is then clarified with a metaphor drawn from ancient eastern writings: the elephant, the rider and the landscape. With these in mind, the various roles that technology can play in helping us to cope with complexity can be considered.

Applying these frameworks, the central thesis of this paper can be advanced: that efforts to manage experienced complexity using IT-based informing systems frequently lead to the unintended consequence of making the decision environment more complex which, in turn, requires the construction of ever-more complicated systems, leading to ever-growing real world complexity, and so forth. This is an example of a cybernetic positive feedback loop—a loop that inevitably produces behaviors that are wild and ultimately unsustainable.

In concluding the paper, ways to address the cybernetic loop of complexity are proposed. In this context, it is important to avoid cures worse than the underlying disease. For all its faults, real world complexity has the undeniable virtue of continuously generating new opportunities. Thus, the solutions proposed include refocusing education such that it emphasizes ways of reacting more productively to experienced complexity—avoiding the twin temptations of either a) locking in familiar patterns of behavior or b) delegating decision-making to ever-more complicated systems. With respect to informing systems development, the solution is to place a greater priority on the adaptability of systems, as opposed to delivering short term gains in efficiency.

---

[1] The paper is based upon a keynote delivered by the author at InSITE/CCISE 2013 in Porto, Portugal and, a week later, at IREPS 2013 in Orlando, Florida.

2. TASK COMPLEXITY DOMAINS

Task complexity is a construct that, in the broadest terms, attempts to explain how the characteristics of a task impact the cognitive demands placed upon the task performer. Despite a number of attempts, e.g., [2][34], no consensus on how it should be defined has been reached. When I studied the literature [18] several years ago, I found 13 distinct definitions.

Over time, my understanding has evolved to the point where I now view complexity as existing in three distinct but interacting domains. As illustrated in Figure 1, these domains consist of: 1) complexity as we experience it, 2) complexity as manifested in our symbolic representation of the strategy used to perform a particular task, commonly referred to as the problem space [3], and 3) complexity manifested in the external environment and in real world behaviors. These are similar, but by no means identical, to the three types of complexity Campbell [2] proposed. I first describe these domains, then consider how they interact.

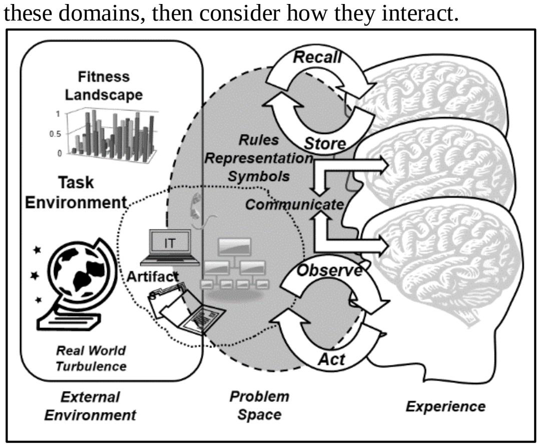

Figure 1: Three Domains of Complexity

2.1 Experienced Complexity

As illustrated in Figure 2, the first domain of complexity relates to the feelings we experience upon encountering a task—particularly an unfamiliar or non-routine task.

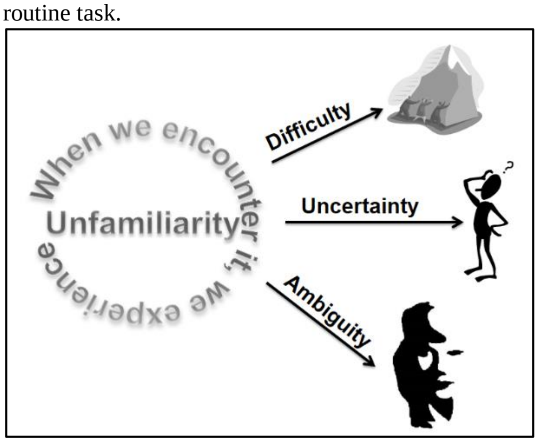

Figure 2: Unfamiliarity produces experienced complexity

Experienced complexity is largely non-symbolic in form, but provides the decision-maker with significant motivation to adopt a strategy to perform the task. It is a response to the task context and, as familiarity with a context grows (e.g., with practice or instruction), the "complexity" that we experience declines.

In the absence of an acceptable strategy, we have evolved a toolbox of non-symbolic mechanisms that allow us to cope with experienced complexity. One of these is a built-in (albeit continually changing) set of preferences, referred to by economists as our utility function [14]. The other is a strong tendency to imitate others, and to cluster together in groups of self-similar peers, referred to as homophily [16]. These largely unconscious behaviors are sometimes derided as being "irrational". To the extent that they are non-symbolic and cannot be justified through a chain of reasoning, this characterization is correct. I will argue later, however, that just because a strategy is irrational does not necessarily mean it is a bad strategy, only that its logic is not immediately apparent.

2.2 Problem Space Complexity

The second type of complexity is the most commonly used. To be determined, it requires a strategy be in place to perform the task, and that the strategy be articulated in symbolic form. At a minimum this requires: a set of symbols that can be used to represent the task, a mechanism for representing relationships between symbols and a set of operators or rules for manipulating these symbols [3].

Once a particular strategy has been represented, often referred to as a problem space, it possesses a number of characteristics not available to task performers operating entirely in non-symbolic mode. These include:

- Symbols can be communicated to other task performers more reliably than non-symbolic feelings
- Symbol sequences can be stored and recalled to overcome working memory limitations.
- Artifacts, such as information systems, can be employed to store, communicate and automate symbol processing.

Collectively, these mechanisms can be used to expand the range of task contexts over which a particular strategy can be employed successfully.

A fully formed problem space shares many of the characteristics of computer program. Thus, it is not surprising that many complexity metrics proposed resemble those developed for program complexity. For example, Campbell's [2] "number of paths" is similar to McCabe's cyclomatic complexity [26]. Wood's [34] references to "amount of knowledge" similarly map to Kolmogorov complexity [25]. In referring to a problem space, I have found the latter of these—interpreted as the minimum description size needed to capture the strategy—to be a good working definition.

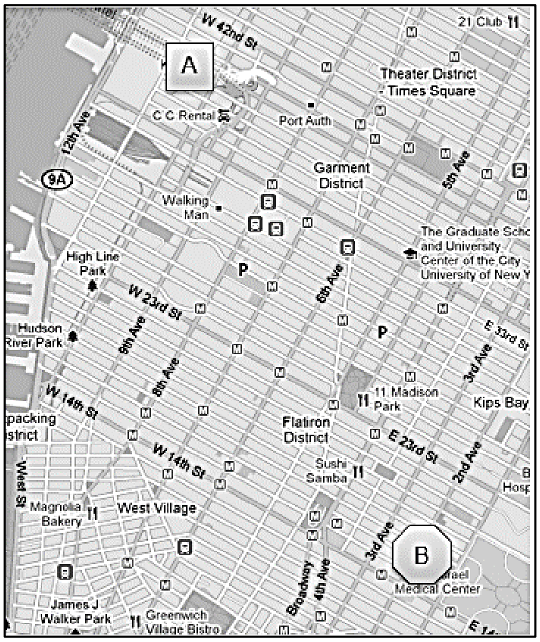

Figure 3: Manhattan street map

To illustrate the relationship between the size of the minimum description and problem space complexity, consider how the length of directions from an arbitrary Point A to an arbitrary Point B might differ for Manhattan (Figure 3) and Boston (Figure 4). Because of the regularity of the grid-based Manhattan streets, most directions in the region displayed can be expressed in terms of a certain number of blocks southwest and a certain number southeast. While many paths conform to these directions, they all get you to the same place.

Figure 4: Boston street map

Lacking the regularity of a grid, Boston directions are necessarily longer and the cost of error is much higher (particularly given the irregular network of one-way streets). Thus, we would describe the problem space for giving directions between arbitrary locations in Boston as being more complicated than the comparable problem space for Manhattan.

To avoid ambiguity, I will henceforth use the term *complicated* when I refer to problem space complexity. A particularly noteworthy aspect of complicatedness is that it tends to grow as we gain experience with a set of task contexts—the opposite of what happens to complexity as we experience it. Naturally, there are exceptions to this rule. From time to time we may be lucky enough to develop insights that allow us to cull unnecessary or incorrect knowledge from our problem space. Such restructuring is rare, however. Individuals who discover such simplifications—Maxwell and Einstein being good examples—frequently earn the well-deserved label of "genius".

Another important way in which problem space complexity differs from experienced complexity relates to the use of artifacts. To use a personal example, when I complete my taxes each year I use a computer running software called *TurboTax*. Because the software guides me through the process of filling out the forms, I find the task less difficult than I did when I used to use pencil and paper. Thus, the software has reduced my experienced complexity. *TurboTax*, however, is a versatile product that can handle many different tax contexts. By bringing it into my personal tax problem space, I have made the resulting problem space more complicated (if I were to endeavor to describe it completely). In a real sense, I have reduced experienced complexity by substituting complication. This is a common impact of employing IT, to be discussed at greater length.

The final way that problem space complexity differs from experienced complexity involves acquiring assistance from others. Here, the analysis parallels that used for artifacts. Getting help from other people can reduce difficulty, uncertainty and ambiguity. Since non-primitive communications depend heavily on the use of symbols, however, however, we must move more and more of the task into the problem space if we want to enable such sharing. In doing so, we make the problem space more complicated.

2.3 Real World Complexity

The final domain of complexity describes the nature and behavior of the environment in which the task is performed. Although well known to researchers of complex adaptive systems, it is not generally applied to task complexity. This lack of application is unfortunate. In fact, I argue that many of the most intriguing aspects of task performance under complexity are driven by this particular complexity domain.

Real world complexity derives from the structure and behavior of the environment in which a task is performed. There are two aspects that are most commonly referenced in this context: *ruggedness* and *turbulence*.

Ruggedness. The term ruggedness draws upon the concept of a fitness landscape, introduced in evolutionary biology [23]. In the context of task complexity, assume that every task context and task strategy can be described in terms of a set of attributes. Then the fitness of a particular strategy N applied in a particular context M might be described by the function:

$$\text{Fitness} = f(\text{context}_M, \text{strategy}_N)$$

The interpretation of fitness is subtle, and is described elsewhere [17]. For our purposes here, it can be treated as a likelihood that when faced with a similar context, an arbitrary task performer would choose to employ the same strategy. That likelihood, in turn, will almost certainly be highly correlated with the desirability of the task outcome. So, at the risk of oversimplifying, I will treat fitness as a measure of a strategy's effectiveness in a particular context. Strategies that exhibit high fitness across a variety of contexts tend to persist and evolve over generations of task performance. Low fitness strategies do not.

The ruggedness of the fitness landscape describes the degree to which the various individual attributes of the fitness function interact in combination, rather than independently of the values of other attributes. I have found the easiest way to explain this is through example. Consider how the structure of the fitness function might differ for the answers on a test versus the ingredients in a recipe, as illustrated in Figure 5. For a typical 60 question test, fitness would reflect the total score and each question would represent an attribute. Each question would be worth a certain number of points, so getting question 12 right will *always* lead to higher fitness when contrasted with getting it wrong.

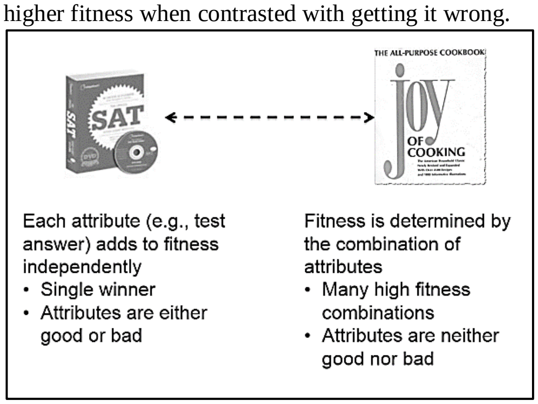

Figure 5: Decomposable versus rugged

For a recipe, we might imagine fitness to be some measure of how good the resulting dish tastes and the attributes would be represented by quantities of individual ingredients and some representation of preparation process. For this landscape, however, there is likely to be no strict relationship between attribute values and fitness. Garlic, for example, is a wonderful addition to some recipes and would doubtless ruin others. Thus, when we thing about rugged landscapes, we must think in terms of combinations of attributes that *fit* together properly—i.e., the relationship between an attribute and fitness is not decomposable.

As the ruggedness of a landscape grows, a number of changes occur:

- The number of local peaks—combinations where any incremental change to an attribute produces a drop in fitness—grows.
- The typical change in fitness associated with incremental changes grows.

An interesting side-effect of increasing ruggedness is that it tends to increase the benefits of imitating the behavior of high fitness nearby neighbors [16], particularly when compared with general expertise.

Generally speaking, ruggedness grows with the number of attributes impacting fitness and, more importantly, the strength of the interaction between them. It is also possible that the behavior of individuals on the landscape will, itself, impact fitness. For example if everyone decides to build the same type of restaurant on the same block, the fitness of that particular strategy will decline rapidly. It is also possible for related systems to impact each other's fitness landscapes, a process referred to as co-evolution in the biological context [23].

One empirical consequence of rugged landscapes that continually adapt is that evidence of fitness—such as wealth of individuals, population of cities, citations to papers—frequently distributes itself according to a power law such as the 80-20 rule [15]. This distribution is largely empirical in origin, although simulations and mathematical models have been used to show why such patterns tend to arise under certain circumstances, such as traffic through network routers [33]. The presence of such distributions necessarily results in considerable inequality of fitness outcome. For example, in an 80-20 distribution roughly half of all fitness is distributed to the top 1% of all entities.

Turbulence. Whereas ruggedness describes the distribution of fitness at a particular point in time, turbulence [12] describes how the state of a system and its structure changes over time. A common continuum used to characterize such dynamics is:

Ordered → Complex → Chaotic [12]

At one extreme, ordered dynamics consists of stable patterns or cycles that are predictable in nature. At the other extreme, chaotic behaviors appear random, even though they may have some underlying structure, such as "strange attractors" [19]. The "complex" intermediate state is the one normally associated with turbulence. Its behavior is characterized as *punctuated equilibrium*, a dynamic where periods of relative order are interspersed with unpredictable jolts or discontinuities during which system behavior changes suddenly and often quite radically. These jolts may be the consequence of forces outside the system being observed ("Black Swans") or as a result of the dynamics within the system ("Grey Swans")[30].The contrast between orderly growth and turbulent growth is shown in Figure 6.

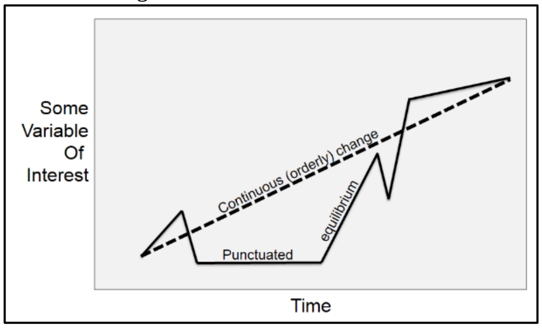

Figure 6: Punctuated equilibrium versus continuous (orderly) change

There are many parallels between turbulence and ruggedness. For example, the same type of power law that is empirically observed in the distribution of fitness is often found in the size of the discontinuities experienced in complex systems [30][16]. Parallels between the presumed sources of the phenomena are also readily discernable, as shown in Figure 7.

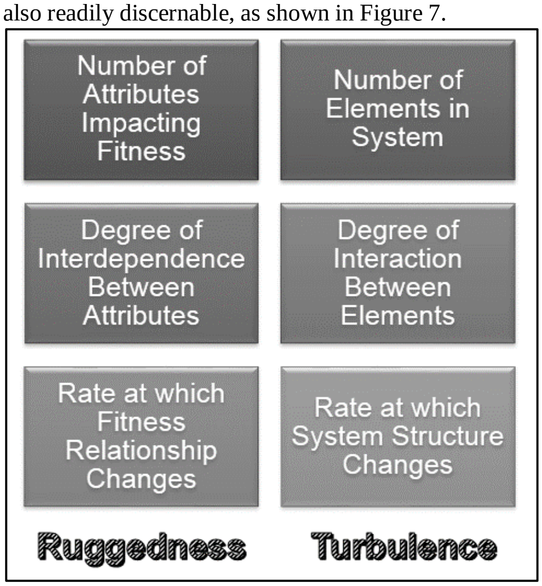

Figure 7: Parallels between ruggedness and turbulence

Because the three domains of complexity that I have described are so different, it is reasonable to ask if it makes sense to consider collectively. The justification for doing so based in the interactions between them. Indeed, as is so often the case with complexity in any form, much is lost when viewing component parts separately. I will try to explain with an analogy.

3. THE ELEPHANT-RIDER METAPHOR

Drawn from ancient eastern philosophy, a number of psychology [20] and management [21] researchers have used the metaphor of a rider (rational thought) atop an elephant (emotions) to describe the way we make decisions. The basic message is that the rider thinks that he or she is in control until the elephant decides otherwise. This applies nicely to complexity provided we add one additional element—the terrain in which the travel is taking place. In my version, shown in Table 1, the elephant represents complexity as experienced, the rider represents the task problem space and the terrain represents the environment.

Table 1: Elephant-Rider Analogies

Elephant-Rider	Complexity
Elephants accept the guidance of riders unless it conflicts with their needs	Where a satisfactory problem space exists for performing a task, task performers will generally be content to apply it.
The rider is in control until the elephant thinks otherwise.	Task performers will abandon a problem space if continuing to apply it becomes too difficult, uncertain or ambiguous.
Riders tire out a lot faster than elephants.	When the complexity we experience from employing a particular problem space is too high, we will abandon it.
When the terrain gets too scary, elephants stampede.	Environmental jolts can cause decision-makers to abandon their problem space and rely on "gut feel".
Elephants like to be in herds.	When the environment causes decision-makers to abandon their problem space, they actively seek to imitate the decisions of others.
When travelling in a group, riders believe they follow the other riders; elephants believe they follow the other elephants.	When pursuing a symbolic strategy, we follow and seek guidance from others pursuing similarly "rational" approaches; when driven by our feelings, we simply follow others who seem to know where they are going.
A lone elephant can do some serious damage; a herd can dramatically alter the terrain.	The behavior of entities in the course of performing their tasks can change how the environment behaves and the underlying fitness landscape function.
Through rugged terrain, it's easier to follow a path that has already been beaten down regardless of how it was made.	As real world complexity increases, we often find ourselves following the practices of other decision-makers without necessarily knowing if their choices were rational or emotional.

What is important about Table 1 is not the specific behaviors described—I freely confess that I am not sufficiently expert in rider-elephant dynamics to know if they are valid, or merely plausible-sounding nonsense. Rather, the analogy is presented to point out the importance of interactions between the different domains of complexity. Specifically, it highlights the fact that well-laid strategies for performing a task tend to be abandoned when experienced complexity grows too large. Sources of experienced complexity include:

1. Working memory capabilities are exceeded. Human beings can attend to only a limited number of conceptual objects at once (5-7 being the most common estimate [27]).
2. Relevant data is not available or cannot be accessed from our long term memory.
3. We are unable to find or communicate with other performers who may have more appropriate problem spaces for accomplishing the task.

In all three of these cases, experience with the task reduces the challenges of task performance. Practice reduces the amount of attention that must paid to performing the task while increasing the size of task-related conceptual objects that we can hold through a process called chunking [32]. The same practice builds links to knowledge that we must retrieve from long term memory, making it easier to access. Finally, with experience we become equipped to handle more concepts.

Real world complexity interferes with our ability to cope with experienced complexity through repetition. The interference is both a consequence of ruggedness (many more possible fitness peaks to consider) and turbulence (the nature of task consequences changes significantly). In essence, real world complexity means we are likely to encounter fewer routine tasks and those tasks are likely to remain the same for a shorter period of time.

4. INFORMING SYSTEMS AND COMPLEXITY

An informing system is a system that is has either evolved or been designed to supply the information required to perform a task or collection of tasks [6]. Very frequently, the construction of these systems involves the use of IT artifacts, although systems that rely purely on person-to-person communication still qualify.

4.1 How Informing Systems Transform Complexity

Informing systems play a crucial role in coping with increasing real world task complexity [18]. I will briefly describe four mechanisms through which such systems—particularly when augmented by IT artifacts—can be used to manage experienced complexity. These are presented in Figure 8.

Proceeding clockwise from the left, the first of these—improved ability to observe and control—is readily illustrated by the implementation of enterprise requirements planning (ERP) systems, which include both tools for acquiring and analyzing data, providing top level managers with a summary view of the organization and mechanisms for implementing routine policies such as purchasing.

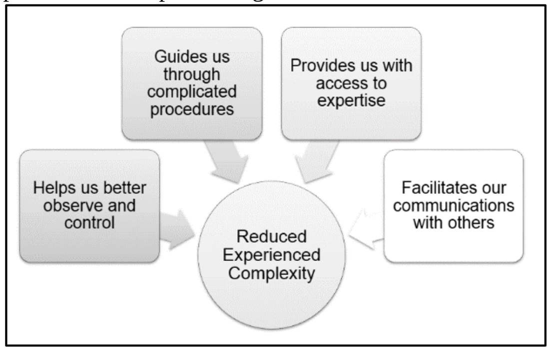

Figure 8: Ways that informing systems reduce experienced complexity

The next two are illustrated by an earlier example I discussed: using *TurboTax* to complete my U.S. income taxes. Conceptually, what I was doing was augmenting my personal problem space to include all or most of the U.S. tax code (estimated to be about 4 million words with an average of more than one change per day [1]), thereby reducing the need for me to: 1) acquire the code in long term memory, 2) apply the relevant elements of the code to my own situation, a working-memory intensive operation, and 3) perform the various computations necessary to achieve the all-important "taxes owed" final value, also demanding considerable working memory. In addition, *TurboTax* provides me with access to expertise, both in the interpretation of the tax rules incorporated in its program and through the capability to engage in live chat with experts.

An example of facilitating communications with others, the recent explosion in social media has dramatically expanded our ability to engage in peer-to-peer exchanges. This represents just one more step in an evolution of the web from broadcasting (e.g., static web pages), to dynamic data-driven applications (emphasizing purposeful exchange of task-relevant content) to peer-to-peer sharing (recreational exchange of arbitrary content).

Broadly speaking, what informing systems and IT artifacts accomplish is reducing experienced complexity by moving it into the symbolic problem space, which is where both IT and communications take place (since they depend on exchange and manipulation of symbols). Using my previous vocabulary, we make the world feel less complex by making it more complicated, then hide the growing complication with technology or division of labor. I grudgingly acquiesce to the 4 million word tax code because an informing system reduces my need to confront it directly. By acquiescing in this manner, we enable further growth in complication.

4.2 Direct Impacts of Increasing Complication

As a general rule—with the notable exception of what economists refer to as "inferior goods"—if you want people to consume more of something, you lower its price. When we build IT-artifact driven informing systems, we are effectively reducing the "cost" of engaging in certain types of activities, making it likely that we will rely on them more heavily in our future task performance, framing our tasks so they take advantage of the capabilities that IT provides. This is illustrated in in Figure 9.

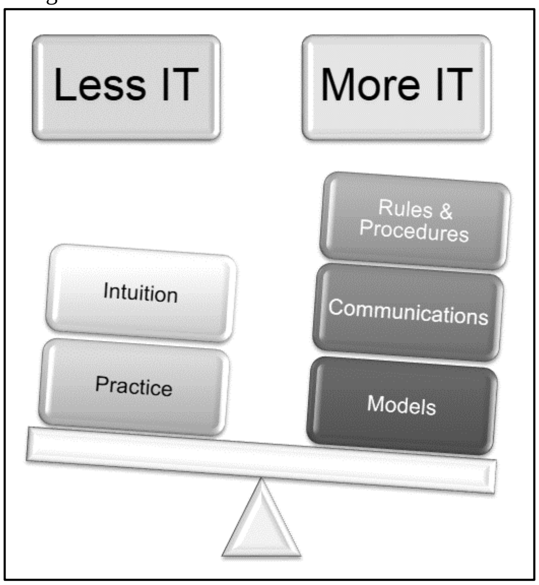

Figure 9: IT and how we frame problems

How we frame problems will, in turn, impact the skills and priorities we set forth in our problem solving and task performance. This is illustrated in Figure 10.

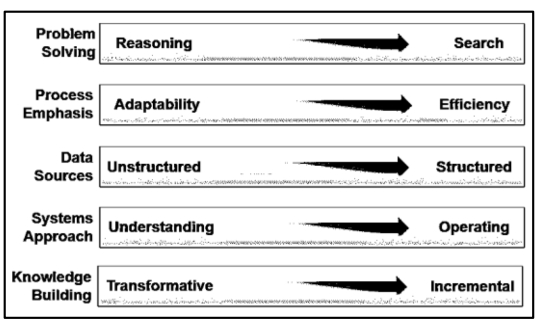

Figure 10: Direct impact of employing IT-artifacts on task performance

The rationale behind the figure is relatively straightforward. If it costs less to search for someone else's solution to a problem, we will have less incentive to reason it out our own. If we rely on systems—such as ERP installations—to manage our processes, it becomes easier to manage for efficiency and harder to change the processes that are in place [12].

The more we rely on IT to supply and process our information, the greater the weight we will place on the type of structured information that is easily processed and interpreted. Today, for example, U.S. high school students apply to more colleges than ever before [22]. Explanations for this are dominated by two factors: the web makes it easier to find out data about colleges and the online "Common Application" makes it easier to apply to many at once. Assuming that these students have not discovered a way to manufacture more time, the obvious conclusion is that they must be relying more heavily on the structured information readily available online (such as *U.S. News and World Report* rankings) than on visits and less structured sources.

Finally, as illustrated by my previous discussion of how I use *TurboTax*, IT has transformed my tax-preparation concern from one of understanding the tax code to one of understanding how to operate the artifact.

The last of the Figure 10 skills, "knowledge building", deals with how we choose to attack problems whose experienced complexity exceeds our capabilities. Prior to IT-enabled informing systems, the choice was often to fail or to radically restructure the problem so as to bring it within our capabilities. For example, the self-evident deficiencies of the Roman numeral system when performing complicated computations required a rethinking of how to represent numbers. When IT-enabled informing is available, however, our ability to add incrementally to already over-complicated problem spaces grows dramatically.

Whether or not these changes are to your taste, it is arguable that they make sense given the ever-expanding capabilities of IT-enabled informing systems. Does it really make sense to spend time memorizing information when it is available 24/7 through your computer and smartphone? Is it so important to reinvent the wheel by reasoning problems out when other people's solutions are so readily found through online search? The danger with this line of thought is that one of the most important ways in which we acquire skills—such as reasoning, judgment and memorization—is through practice [32]. If we continually use IT as a crutch, where will those skills be when we *really* need them?

5. EXAMPLE: THE EVOLVING TABLET

Before proceeding to the topic of how IT-enabled informing systems can impact real world complexity, it is useful to consider a brief example centering upon how our conception of a "tablet" has evolved over time. In presenting this material, I have made no effort to be historically complete. Instead, my hope is that the series of snapshots that follow will help to clarify the concepts of fitness and turbulence and the role that IT can play in impacting them.

5.1 Tablets "Before the Current Era"

Tablets have been around for a very long time. For our purposes, I will define a tablet as being an artifact that you can write or draw upon directly. Broadly speaking—and I mean very broadly—tablets have typically had two functions: to broadcast information and to act as a local storage repository for record keeping or note taking. As a result, they have often come in two form factors. As shown in Figure 11, many thousands of years ago these might have involved stone (large form factor, particularly useful for broadcasting) or wax/clay (small form factor, particularly useful for record keeping).

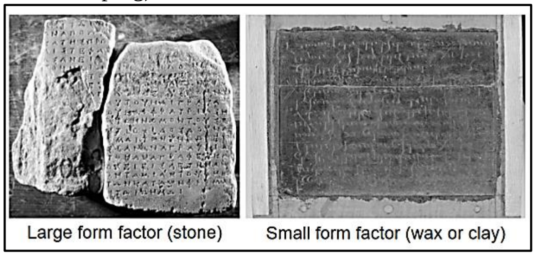

Figure 11: Typical tablets in the BCE time frame

5.2 Slate-based Tablets in Education
In the U.S., during the late 1700s and 1800s, use of the chalk-on-slate tablet format became widespread, as shown in Figure 12. Although not a new technology, its fitness was increased by the social movement towards universal education. Particularly during the beginning of that period, paper was very costly—according to a helpful guide/interpreter that I encountered at Colonial Williamsburg, a single sheet could cost the equivalent of 1-2 hours of a typical laborer's time. Hence the ability to employ the easily erased small form factor version was highly beneficial in the classroom. And, of course, the large form factor was well suited to broadcasting information to the classroom.

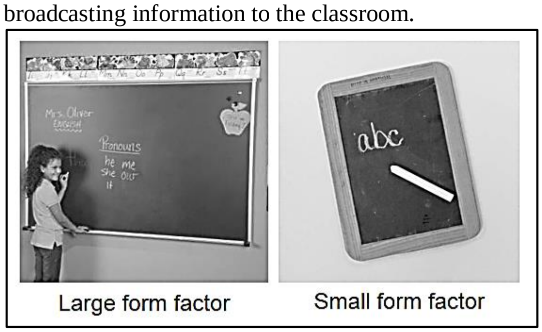

Figure 12: Examples of slate on chalk tablets

5.3 Paper-based Tablets
By the middle of the 20[th] century, a number of factors increased the relative fitness of paper pad-based tablets that reduced the need to erase work. First, the real price of paper had declined by many orders of magnitude. Second, a number of technologies had been developed that increased the flexibility of paper, allowing it to be copied (e.g., carbon paper, mimeograph stencils, ditto sheets and, most important, xerography) and transmitted nearly instantly (e.g., fax). As shown in Figure 13, a number of new variants also emerged during the period, such as whiteboard—which gradually supplanted the slate tablet and its attendant chalk dust—and a particularly nifty new tiny form factor, the *Post-It Note*.

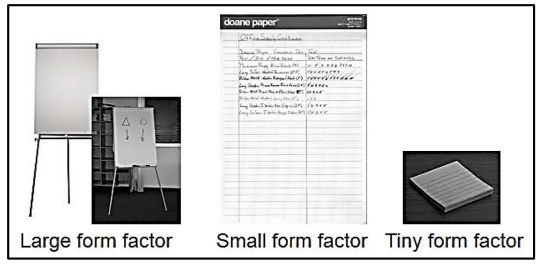

Figure 13: Popular tablets of the late 20th century

5.4 IT-Enabled Tablets
By the very end of the 20[th] century, the tablet had become infused with technology. With respect to the large form factor, devices such as the digital whiteboard emerged. Where the greatest action occurred, however, was in the small and tiny form factors, with some representative examples presented in Figure 14.

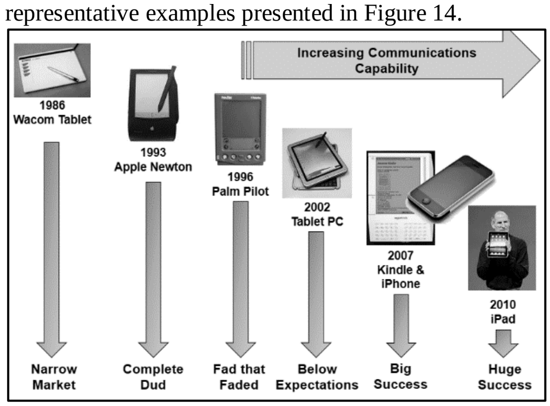

Figure 14: Steps in the evolution of IT enabled tablets

There are two important observations that can be made about the progression of artifacts shown in Figure 14. The first is the apparent ruggedness of the landscape. Indeed, two of the major industry leaders—Apple and Microsoft—had disappointing reactions to their first entries (the Newton and the Tablet PC) despite investing large amounts into their development and marketing. 2007 proved to be the year of breakthrough with the introduction of the iPhone (tiny form factor evolved from the iPod Touch) and Amazon's Kindle (small form factor—although it barely qualifies as a tablet by my definition, since it originally did not allow direct writing or drawing, but at least it looked like a tablet and was an evolutionary precursor to many later designs) What added immeasurably to the fitness of both these entries was the fact that they incorporated communications technologies (wifi and cellular) that had become ubiquitous. Here, they had a significant advantage over earlier portable digital assistants (PDAs) that, at best, supported wifi at a time when it was much less commonly available. The evolution into the tablet/smartphone era fundamentally changed the fitness landscape by making communications capability a critical attribute of the tablet fitness function and making the relationship between other attributes—such as size—and fitness more ambiguous.

6. THE CYBERNETIC LOOP

Part of what makes the potential impact of IT-enabled informing systems so concerning is the associated indirect impacts. The source of these effects can be framed as a positive feedback loop that continually increases real world complexity.

6.1 The Complexity Cybernetic Loop

Consider the following sequence:

1. High experienced complexity causes us employ IT-enabled informing systems to reduce difficulty.
2. These systems allow us to introduce and manage more complicated processes that communicate to gather more information and allow us to react more quickly.
3. A side effect of these advanced systems is to increase the number of individuals and entities connected into our systems, the degree to which they can interact with each, and the speed at which they can react to each other's actions.
4. Number of elements, level of interconnection and speed of reaction are the fundamental contributors to turbulence, which therefore tends to grow.
5. Increased turbulence leads to greater uncertainty and ambiguity. Experienced complexity grows, and individuals reach out for solutions—with IT-enabled informing systems being the first choice.

With that, the process repeats, as illustrated in Figure 15.

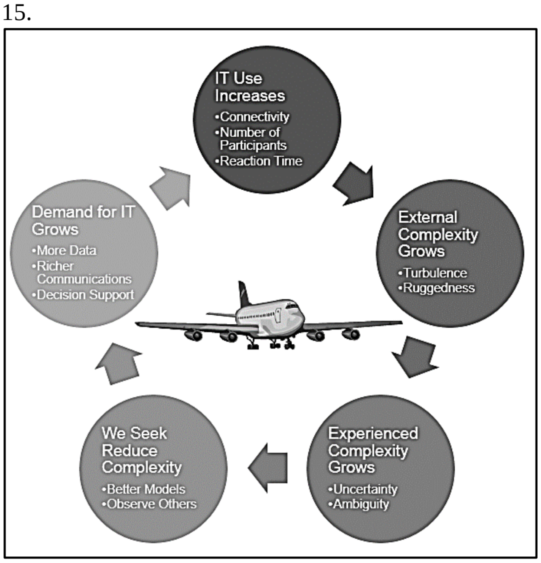

Figure 15: Cybernetic complexity loop

6.2 Airline Reservation Systems

The jet pictured in the middle of Figure 15 is intended to remind us of one of the earliest examples of such an IT-enabled loop: the airline reservation systems [15]. In 1978, the U.S. deregulated its airline industry, allowing prices and routes to be set more competitively. To handle these new freedoms, the airline reservation systems—that had previously served only a supporting function in booking—became critical components of the competitive strategy of those airlines that owned their own systems (particularly American and United Airlines). Using their ability to change prices and control information flow to travel agents, these systems were used to bring competing airlines, such as Braniff and Frontier, to their knees. The resulting turbulence engendered by these rapid shifts in price, fare structure and routes caused what had previously been a relatively stable industry to become exceptionally volatile. And as turbulence increased, so did the need for ever-increasing reservation system functionality.

6.3 Effects the Loop

The indirect impact of the cybernetic complexity loop can be framed in terms of what it means to live in a world of ever-increasing real world complexity. Or, more specifically, growing ruggedness and turbulence. We now consider a number of these effects identifying, where possible, the role that IT has played or is currently playing.

Increased entrenchment. Where ruggedness is high, we expect to observe the tendency to hold on to a particular peak rather than risk a misstep that could lead to a sharp drop in fitness, a form of expert entrenchment [8]. Because ruggedness means fitness depends on attributes acting on fitness in combination, rather than individually, competing perspectives often find compromise unattractive, since the "middle ground" between two local peaks is often a valley.

As real world complexity increases entrenchment, we can increasingly expect to see phenomena such as Christensen's "innovator's dilemma" [5], illustrated in Figure 16. The process begins when an organization develops an innovation with a particular price-capabilities combination that is a good fit with a specific group of customers (i.e., occupies a local fitness peak). As shown in the oval, over time the price-performance of this innovation improves as the organization stays focused on its core customers.

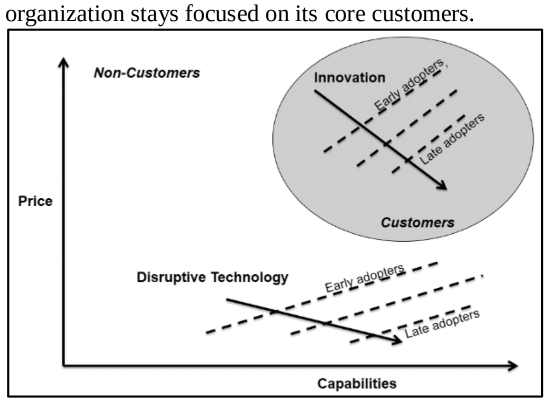

Figure 16: Innovator's Dilemma

Meanwhile, some other innovation—referred to as a "disruptive technology"—is introduced serving a very different set of customers. In the figure, that new technology starts at a substantially lower price point and similarly reduced capabilities, rendering it unsuitable for the satisfied customers of the original innovation. For this reason, the original firm remains entrenched in its view that its innovation has the higher fitness. Over time, however, the disruptive technology's capabilities improve to the point where they overlap those of the original innovation. At that point, customers of the original firm jump to the new technology en masse, leading to the type of sudden shift that typifies turbulent dynamics. Christensen cites many examples of this process playing out—including the integrated steel industry, disk drives and mini-computers [5].

IT systems are particularly prone to producing entrenchment by virtue of the fact that they codify a procedure in a manner that makes it difficult to change [12]. Thus, systems designed to enhance the efficiency of a particular strategy for task performance—such as an ERP system or workflow manager—can increase an organization's reluctance to make significant changes that require restructuring or abandoning existing IT-supported processes.

In addition, informing systems such as social networks and online chat-based special interest groups can further enable such entrenchment by allowing individuals sharing the same perspectives to communicate with each other, reinforcing beliefs, even when geographically or socially dispersed.

Increased imitation. In simulation studies, such as that shown in Figure 17 [16], I have demonstrated that when ruggedness is high, it is often better to imitate self-similar peers than rely on general experts. The X-axis of the chart represents the complexity of an NK landscape [23], where 0 means each of the 10 attributes simulated contributes to complexity independently (fully ordered) and 9 means that nothing short of knowing the precise combination of all 10 attributes tells you anything about the resulting fitness (maximally rugged). The Y-axis is the number of steps each entity takes to reach a local fitness peak. The graph shows that *as complexity grows*, we are better off being guided by mimicking nearby neighbors (the technique employed by both the "imitate" and "goal" strategies) than taking guidance from an "expert" who derives simple single-attribute rules by looking at the entire population of entities on the landscape, whose performance converges with that of random guessing.

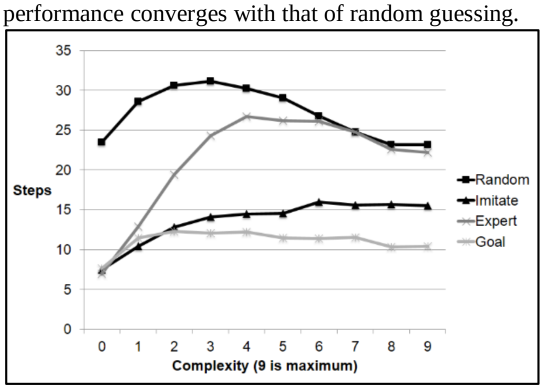

Figure 17: Simulation of steps required to reach a local peak as complexity grows [16]

IT, particularly social networking, provides us with numerous tools for observing and imitating the behavior of others. To name just a few:

- *Facebook* allows us to express our likes and dislikes and see those of others.
- *Twitter* allows us to catalog our day's activities and express opinions in short bursts to our followers.
- *LinkedIn* allows us to join groups with individuals having common interests.
- *Amazon* recommends products based upon what other people with similar interests have bought; Netflix does the same for video content.
- *YouTube* tells us how many people have watched a particular video.

I recently attended a presentation of a Hong Kong-based startup called *Viss* that takes the process a step further, allowing users to photograph themselves and then tag where they acquired each item of clothing, thereby permitting other users to duplicate their style.

In the future, IT may enable even easier imitation. Declines in the cost of 3D printing may allow us to manufacture products in the home or office economically. Technologies such as Google Glass may make it easier to identify products and like-minded people without even the need to pull out a phone.

Growing inequality. I have already commented on the empirical tendency of real world complexity to produce power law distributions of fitness-related outcomes. Where a power law is present, inequalities in the distribution of outcomes across a population will tend to be large. And, in fact, evidence suggests that inequality in income has grown over the past decade in many developing countries [10].

In addition to impacting inequality of outcome through impacting real world complexity, IT can specifically foster in inequality through the creation of an ever-growing digital divide. The reasoning here is that those individuals most able to cope with increasing real world complexity will be those who are effective in offloading their tasks to the symbolic world of the IT-enhanced problem space. Those individuals who are not capable of harnessing technology in this way, either through lack of access or lack of training, will then be less able to contribute productively to the workforce. As such, their relative income can be expected to shrink as we become more reliant on technology.

Increasingly complicated systems. As we have already seen, IT can often be used to enable complicated strategies without overwhelming task participants. Regardless of how simple a platform or application seems when it is introduced, it tends to grow more and more complicated as it evolves, since it is easier to add functionality incrementally than to completely restructure an IT artifact. All you need to do is look at how applications software, ERP systems and popular operating systems have evolved to see this trend in action.

The growth of inequality and turbulence that attends increasing real world complexity has also encouraged organizations and policy-makers to attempt to enforce specific behaviors and outcomes though the use of policies. I have already mentioned the U.S. tax code, which has grown to be hugely complicated; to a great extent, this size is a result of a desire to maintain progressivity (i.e., greater equality of outcomes) and to provide relief to certain favored interests or causes. Other U.S. examples include the 2011 Dodd-Frank legislation—2319 pages that specify what regulations *need to be created*—intended to reduce turbulence in financial markets, and the 906 page 2010 *Patient Protection and Affordable Care Act* (a.k.a. Obamacare)—thus far generating between 10,000 and 40,000 pages of regulations, depending upon whom you ask [24]—that was established to provide more universal access to healthcare in the U.S.

You can make a plausible argument that these page counts alone could not have been achieved without IT-enabled informing systems—consider what would be required to assemble, typeset, print and distribute them manually. Beyond that, however, these complicated systems of regulations assume—either explicitly or implicitly—that IT will be used to manage workflow and transactions. The U.S. Internal Revenue Service, for example, is increasingly pressing or requiring taxpayers to prepare and submit their returns electronically rather than on paper, both to reduce costs and error-rates. Dodd-Frank specifies the creation and use of various clearing houses, nearly all of which are going to be electronic in nature. Without the ability of databases to track transactions and patterns, it would be impossible for an already complicated system of financial checks and balances to function. The PPACA mandates the use of electronic health records in order to make the system function more efficiently.

Shortened time horizons. Turbulence leads to a type of uncertainty more serious than that of simple volatility. With pure volatility, you can have confidence that the system will ultimately return to past behavior. Turbulence in an adaptive system offers no such assurance, since changes within the system can fundamentally alter its future behavior. Uncertainty driven by turbulence should therefore cause us to discount the future more heavily and could easily make it more difficult for us to identify with our future selves. This is illustrated in Figure 18, where the widening of the cone indicates the range of future possibilities.

On a global level, the effects of shortened time horizons can already be observed in a number of areas, particularly in the industrialized world. For example, government debt has grown significantly as a percentage of GDP among developed countries in recent years [9]. Similarly, birth rates are well below replacement in most European countries, China and Japan [15]. What this means is that we are currently placing substantial demands on future generations that could well be much smaller than our own, increasing the per-capita burden on our children dramatically.

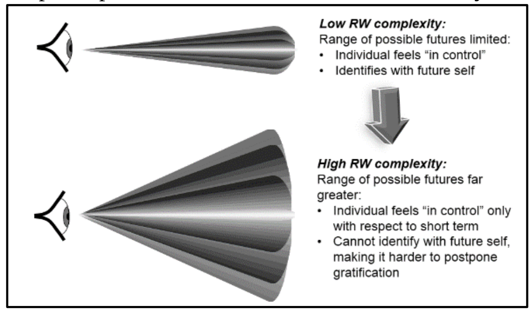

Figure 18: Impact of real world complexity on time horizons

Naturally, the impact of IT-enabled informing systems on real world complexity is only one factor contributing to our time horizons. Nevertheless, it uniquely contributes in two ways. First, IT frequently serves to reduce the time required to accomplish specific activities; on a global scale, this has the effect of compressing the pace of time; in effect, it brings the future closer. Second, IT—along with biotechnology—is one of the areas where the pace of transformational change is greatest. Just over a decade ago, for example, I did not have a wireless network, a smart phone, HDTV, Netflix, a tablet, Skype, text messaging or any type of social presence on the web. The particular challenge this type of change produces is one of imagining what your future self will be like.

Just over a century ago, when similar transformational changes were occurring in the electro-mechanical sphere (e.g., proliferation of telephones, automobiles, air travel, refrigeration, etc.), many individuals held to a comforting "Idea of Progress" [11] that encouraged the shared certainty that the future would be better. Today, an IT-enabled media informing system brings concerns about the environment, acts of terror and a variety of other dire prognostications—often encouraged by entrenched special interests—to our homes and desktops. Under such a continuous barrage, it is hard not to discount the future.

6.4 Summary of Indirect Effects

The just discussed indirect effects of the cybernetic complexity loop are summarized in Figure 19. The narrow nature and sharp drop-offs around rugged peaks mean that we tend to become focused on behaviors around fitness peaks and are less interested in general principles. That same drop-off makes the most successful of us more inclined to preserve what we have rather than expanding our horizons; the growing inequalities fostered by real world complexity make the rest of us want to spread the fitness out more equitably, rather than simply ensuring equal opportunity. Sharp peaks and turbulence make it safer to imitate than to explore new paths, so our innovation becomes more incremental. Finally, the uncertainty and inability to identify with our future selves shrinks our time horizons, with all the attendant problems that brings.

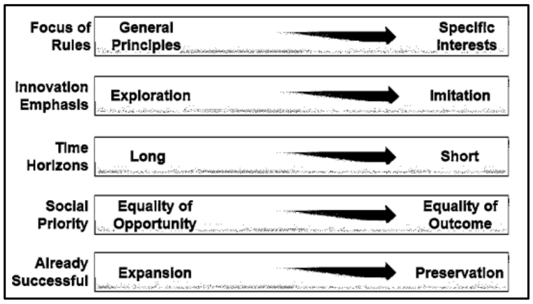

Figure 19: Summary of indirect effects

7. THE BETTER MOUSETRAP

Particularly in light of the previous section, it would be easy to imagine that I suppose real world complexity to be a fundamentally "bad" thing. Such a conclusion would be premature. While I think it would be a mistake to ignore its negative effects, it also has some very redeeming qualities. I will first discuss these, then turn to the question of how we might try to reduce some of its less desirable side-effects.

7.1 The Bright Side of Real World Complexity

For all the challenges it presents, real world complexity has two characteristics that are well worth preserving:

1. It fosters opportunity
2. It encourages adaptability

With respect to the first of these, turbulence tends to prevent entrenched participants perched on a local fitness peak from remaining entrenched very long. Embodying Schumpeter's "creative destruction" and Christensen's earlier mentioned "innovator's dilemma", the discontinuities that accompany real world complexity tend to be hard on established organizations. For example, nearly half the companies on the *Fortune 500* in 1995 were no longer included ten years later [29].

The increasing ruggedness engendered by real world complexity also produces many combinations of attributes whose fitness is untested. As demonstrated by the tablet example, evolution of technology and broader penetration of existing technologies (such as cellular and wifi networks) creates new sweet spots—the world was not ready for Microsoft's Tablet PC in 2002 [13] but it was more than ready for Apple's iPad in 2010.

The other benefit of real world complexity is that it tends to reward adaptability. Today, it is fashionable to tout the benefits of sustainability—and to the extent the term is interpreted as meaning not being wasteful and not polluting the environment unnecessarily or in ways that cause permanent harm, I am certainly a fan. If, however, we interpret it as engaging only in activities that can continue indefinitely, we should be careful what we wish for. Elsewhere [15] I have noted that the period of western civilization that achieved the most perfect balance—i.e., was the most "sustainable"—is referred to as the Dark Ages. Historians have argued that the highly structured system in place at the time could have lasted for millennia, had it not been interrupted by the "discontinuity" known as the Black Death. If they are correct, then those of us who are of European descent (and are alive today) owe a great debt to a plague that wiped out nearly half the population of Europe.

The problem with stability—much like the problem of relying too heavily on systems that force us to manage our processes in a particular way—is that the longer it goes on, the less ready we are to adapt to what happens when a Black Swan (e.g., plague, asteroid, innovative new technology) comes along and disrupts the system. When real world complexity is high, we are less likely ever to get that comfortable. The question then becomes how to keep the associated discomfort to a manageable level, such that we do not become slaves to imitation unwilling to think about the future.

7.2 Rethinking Education

Recalling the earlier Figure 10, reliance on IT-enabled informing systems tends to encourage certain modes of thinking in preference to others—search in preference to reasoning, incremental in preference to transformative, operating in preference to understanding, and so forth. The problem these tendencies present is that they tend to form their own cybernetic loop. The less we practice building deep understanding (because pre-made solutions are so easy to find on the web), the harder it becomes to acquire that understanding, making us even more dependent upon technology, causing us to practice even less, and so forth. Eventually, we end up becoming vastly different from our forebears, and less capable of coping when anything goes wrong with our technologies.

IT-enabled informing systems are not going away—nor would I want them to—so we must accept the fact that they will continue to be used where it makes sense to do so, and often even where it does not… What this suggests to me is that we ought to look at Figures 10 and 19 and decide which skills are too valuable to be lost, then focus on having students practice them throughout their education, so that they will be able to draw upon them later in life at a time when they need them. I will provide some simple examples of what I mean, although this is far too broad a subject to be covered in a short paper or presentation.

Reasoning versus search. I often hear faculty members and K-12 teachers complaining about how students would prefer to search the Internet than study the textbook. Personally, I do not see much difference between looking something up on the Internet and looking it up in an impossibly large textbook. It seems that many of us have become obsessed with the amount of content that we "cover" without worrying about whether it has been sufficiently well practiced so that it can be applied conceptually.

The solution I see to this problem is to dramatically reduce the amount of content we expect students to encounter but to ask questions continually that require our students to think deeply about what they have covered. It also implies that we pay far more attention to the processes students employ in their studies, as opposed to the answers they attain. Sadly, that type of attention is far more demanding of the teacher's or instructor's time, which will make selling it a challenge.

I have at least one relative who confessed to me that she relied entirely on *SparkNotes*™ study guides for all her assigned literature reading in college, doing quite well in the process. I would argue that when outlines are effective substitutes for literature, we are ensuring our students are well-prepared for consuming the type of superficial results returned by a typical web search.

Adaptability versus efficiency. As per the above *SparkNotes*™ anecdote, given the same type of material over and over again, students will find "efficient" ways to complete it; using study guides actually being one of the more benign. Moreover, we seem to be encouraging students to determine their preferred "learning style" and to demand it—at least that is what I infer from all the students who have approached me to alert me to the fact that they are "visual learners".

What I would propose is that if we wish to encourage adaptability in our students, we need to ensure that they are presented with a portfolio of instructional approaches—spending far less time on traditional lecture methods. The more often they are confronted with approaches that require them to adapt, the more adaptable they will become.

Unstructured versus structured information. I have recently started using discussion cases as the central pedagogy in my undergraduate capstone class. While I am reasonably satisfied with the effort my students put forth, it is clear to me that many of them struggle with the notion that the situations presented are far from clear cut, the information required is not well bounded and there is probably not a discernibly "right" answer. We need to provide students with a lot more opportunities to practice this type of thinking, the earlier the better. Handling this type of information will help them cope with the uncertainty that they will encounter in the complex real world. The better they can cope with these ill-structured settings, the less prone they will be to overreacting to turbulence and falling victim to "animal spirits".

Understanding versus operating. As an instructor in the information systems area, I am frequently startled by the degree to which I see students—and colleagues—proficiently operating software that they do not fully understand. I should not be surprised; I confessed my own guilt in this matter earlier in discussing how I prepared my taxes. Nevertheless, we need to think about the how we allow students to use tools prior to developing a deep understanding of what the tool does. Graphing calculators, spell/grammar checkers, tutorial software and a host of other tools can all be operated to provide the appearance of understanding without the reality. The same can be said of the sophisticated statistical software frequently employed by academic researchers in the course of their data analysis.

I see no easy solution to the problem of encouraging understanding instead of operating. Although forcing students to develop manual proficiency prior to allowing them to use tools might work in some cases, that seems a bit draconian in today's age and, I suspect, using tools may help many individuals acquire and reinforce their understanding. A better solution may be to design practice problems that require active involvement of the problem solver in addition to the use of tools. Requiring student to explain what they did without reference to the tool may also be useful. The first step, however, is to recognize that the problem exists.

Transformative versus incremental. Much of what we teach involves building upon existing knowledge, and this is often the most sensible approach to a problem. This type of approach, however, nearly always leads to more complicated systems—largely as a matter of definition. If we want to avoid a world that only becomes more complicated, we need to encourage people to consider the alternative of examining problems from a fresh perspective.

One of my favorite educational stories involves the teaching approach employed by biologist Louis Agassiz who would take a new student, place a jar with a fish in front of him and give the instructions:

> Find out what you think you can without damaging the specimen; when I think you have done the work I will question you [7](p. 125).

For several days, Agassiz would return, ask a question or two, and provide little more feedback than "that is not right". In the process, the student would feel impelled to continue to reconsider that fish. Eventually, Agassiz might add more specimens, again providing little or no guidance regarding what he expected. Towards the end of this process, one of these students described in the essay observed something subtly different about one of the specimens. On pointing it out to Agassiz, the professor replied: "Boy, there are now two of us who know that" (p. 126).

If we are to avoid systems that become so complicated that they cease to function reliably, we need more people who can restructure existing knowledge and invent new knowledge as a consequence of studying the fish without preconception.

Indirect effects. With respect to the indirect effects of real world complexity, such as those listed in

Figure 19, I have no specific recommendations. What I would observe, however, is that the complexity cycle tends to become more severe when we overreact to real world complexity. Through better understanding of the typical behaviors of complex systems, we may avoid becoming so alarmed by their side-effects, such as widely divergent fitness outcomes and entrenchment, and recognize that subsequent turbulence will do a great deal towards sorting them out. We may also recognize our own limitations in attempting to compensate by introducing excessively complicated systems to combat them. If we become less alarmed, and take some comfort in the self-correcting tendencies of such systems, we may find ourselves a bit less distressed about the future, and a great deal happier.

7.3 Building Better Informing Systems

In thinking about how the model of complexity that I have presented here might be applied to informing system design, there are two general objectives that make sense to me:

1. Build systems that are robust enough to survive real world complexity.
2. Build systems that do not unduly accelerate the cybernetic complexity loop.

My strong suspicion is that the two objectives are closely related, as I will later explain, but I will nevertheless briefly consider each separately.

Building robust systems. There are two general approaches employed by living systems to survive the frequent discontinuities presented by nature: adaptability and diversity. We should continually be thinking about how we might imbue our systems with these properties.

With respect to adaptability, a key goal is to avoid systems that are too complicated. Complicated informing systems tend to achieve efficiencies in performing the task that they are designed for but they also tend to be brittle when the task changes significantly. That brittleness either leads to system abandonment or, perhaps more often, entrenched business processes that refuse to change in response to fundamental changes in the competitive environment. I see this as one of the greatest dangers associated with massive ERP systems.

In his seminal *Sciences of the Artificial* [28], Herbert Simon establishes that a system can be made significantly less complicated to construct if it is built of loosely coupled components, as opposed to being designed as a single artifact. There are currently numerous trends in software architecture that suggest that we are already rapidly moving down this path. For example:

- Virtualization is reducing the degree of coupling between our systems and the associated hardware platform. It has become the de facto approach to cloud computing.
- Web service architectures are decoupling and clarifying the interactions between application modules.
- Component-based development, web mash-ups and plug-in platform architectures—such as seen in web browsers—are encouraging us to assemble applications from a stock of common components.
- Mobile and desktop apps that are intended to be useful for a small set of task contexts are rapidly overtaking comprehensive (and complicated) desktop applications in popularity and use.

While these and many other examples suggest that we have the toolkits necessary to build less complicated systems, it is less evident that we have: a) placed sufficient priority on achieving adaptability in our designs, and b) have achieved sufficient wisdom to establish the most suitable functional boundaries for our components. When we install a large system—such as an ERP—do we spend enough time worrying about how its presence will affect our behavior when the world changes? Does it make sense that the "premium" version of an operating system, software product or an app is just the "basic" version with additional features grafted on? We need to spend more time thinking about complexity if we are going to come up with answers to questions like these.

It is a little trickier to imagine how diversity can be incorporated into a systems design philosophy. I can think of some possibilities, however. For example:

- Solicit design concepts for a system from diverse groups of individuals, specifying as few requirements (that force the design in a particular direction) as possible. When diverse groups are pressed to engage in transformative thinking, a considerable variety of alternatives should result.
- Have diverse groups participate in development. The open source movement embodies this approach, based on the reasonable assumption that a project is improved by having many different eyes inspecting it.
- Build systems that require active involvement of the user, rather than simply directing the user. Shoshana Zuboff refers to this process as *informating* a job [35]. The benefits of this philosophy extend beyond the direct impact of enriching work. When a diverse group of users operates a system, it will be used in a diverse set of ways likely to impact its evolving design and make it more resilient to environmental changes.
- When investing in systems development, employ a portfolio approach that ensures that some resources flow to higher risk projects with long time horizons [4]. Typically, when short term and long term projects are funded from the same pot, the more we perform comparative analysis, the more likely we are to choose the short term option—since turbulence and ruggedness make any long term analysis we perform speculative at best. Requiring that a certain amount of our attention and resources be directed solely to our long term prospects is the best remedy for this.

Nassim Taleb recently published a book *Antifragile* [31] where he observed that living systems frequently exhibit the property of growing stronger and better when subjected to stress. While the means by which we might incorporate that antifragility into our systems is far from obvious, it is precisely the type of objective needed if we are to cope with complexity successfully.

Building systems that do unduly real world increase complexity. Since we have much to learn about the causes of turbulence and ruggedness, it is hard to provide a formula for keeping the growth of real world complexity to a manageable level. One factor that often lead to greater discontinuities is the buildup of pressure, analogous to the plate pressure that produces earthquakes (the magnitude of which also happens to be distributed according to a power law). To the extent that the pressure analogy is valid, we could expect any system that locks in patterns of behavior for a long time will tend to produce a huge disruption when it is subjected to too much stress. The "innovator's dilemma" provides an illustration of this process, as do a number of recent disruptions in various global financial systems. Systems built to adapt rapidly, on the other hand, will prevent such pressure from building up. Thus, by building adaptable systems we may prevent—or at least reduce the magnitude and frequency of—the grey swans (such as bubbles) that today threaten entire industries and economies. Similarly, if we are confident that our systems can withstand the turbulence of the competitive environment, we may be less inclined to overreact—thus becoming part of the problem—when such inevitable turbulence is actually encountered.

8. CONCLUSIONS

Historically, different researchers have perceived complexity in different ways. Researchers of complex adaptive system have framed it in terms of system behavior. Abstract theorists and computer scientists have framed it in symbolic terms. Behavioral psychologists and researchers studying job enrichment have viewed it from the perspective of the way it makes us feel. What I have argued here can be boiled down to two simple propositions framed using the metaphor introduced in the paper:

1. It makes about as much sense to treat the symbolic, experienced and real world perspectives on complexity independently as it would be to attempt to explain the behavior of a rider on an elephant in unfamiliar terrain by limiting our vision to either the particular rider, the particular elephant or the particular terrain.
2. The behaviors of a rider on an elephant cannot help but change the terrain through which they traverse and, as that terrain is altered, so will be the behaviors of the elephant and its rider. When a lot of riders talking to each other and elephants following each other are involved, these changes will be larger and will occur faster. Since these changes take the form of a cybernetic loop, it is futile to attempt to ascertain what causes what. But that lack of clear one-way causality does not mean we can or should ignore what is happening in the process.

I have also argued that the complexity cycle offers opportunities as well as perils. Through better understanding complexity, we may be able design approaches to education and the construction of informing systems that help us to cope with complexity and make the world a little less scary.

9. REFERENCES

[1] Associated Press (2013), Report: U.S. tax code is longer than the Bible - without good news, 9 January.

[2] Campbell, D.J. (1988), Task complexity: A review and analysis. *Academy of Management Review, 13(1)*, 40-52.

[3] Card, S.K., Moran, T.P. & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale NJ: Earlbaum.

[4] Cash, J., McFarlan, F.W., and McKenney, J. (1988), *Corporate information systems management*, Homewood, IL: Irwin.

[5] Christensen, C.M. (1997), *The innovator's dilemma*, Boston, MA: Harvard Business School Press.

[6] Cohen, E. (1999). Reconceptualizing information systems as a field of informing science: From ugly duckling to swan, *Journal of Computing and Information Technology*, 7(3), 213-219.

[7] Cooper, L. (1994), Louis Agassiz as a teacher, In Barnes, L.B., Christensen, C.R. and Hansen, A.J. (Eds.) *Teaching and the case method, third edition*, Boston, MA: HBS Press, 125-128.

[8] Dane, E. (2010), Reconsidering the trade-off between expertise and flexibility: A cognitive entrenchment perspective, *Academy of Management Review*, 35(4), 579-603.

[9] Financial Data Team of the Development Data Group (2013), *International debt statistics 2013*, Washington, D.C.: The World Bank.

[10] Gangl, M. (2008), A longitudinal perspective on income inequality in the United States and Europe, *Focus, 26(1)*, 33-38.

[11] Gill, R.T. (1997), *Posterity lost: Progress, ideology and the decline of the American family*, Lantham, MD: Rowan & Littlefield.

[12] Gill, T.G. (1995), High-Tech hidebound: Case studies of information technologies that inhibit organizational learning, *Accounting, Management and Information Technology, 5(1)*. 41-60.

[13] Gill, T.G. (2007), Using the Tablet PC for instruction". *Decision Sciences Journal of Innovative Education. 5(1)*, 183-190.

[14] Gill, T.G. (2008), A Psychologically Plausible Goal-Based Utility Function. *Informing Science, 11*, 227-252.

[15] Gill, T.G. (2010), *Informing business: research and education on a rugged landscape*, Santa Rosa, CA: Informing Science Press.

[16] Gill, T.G. (2012), Informing on a rugged landscape: Homophily versus expertise. *Informing Science*, *9*. 49-91.

[17] Gill, T.G. & Hevner, A. (2011), A fitness-utility model for design science research, DESRIST 2011.

[18] Gill, T.G. & Hicks, R. (2006), Task complexity and the informing sciences: A synthesis, *Informing Science*, *9*. 1-30.

[19] Gleick, J. (1988), *Chaos*, New York, NY: Penguin Books USA.

[20] Haidt, J. (2006), *The happiness hypothesis: Finding modern truths in ancient wisdom.* New York, NY: Basic Books.

[21] Heath, C. and Heath, D. (2010), *Switch*. New York, NY: Broadway Books.

[22] Hopkins, K. (2011), Study: More students apply to more colleges, *U.S. News and World Report*. 20 October 2011. Retrieved from: http://www.usnews.com/education/best-colleges/articles/2011/10/20/study-more-students-apply-to-more-colleges

[23] Kauffman, S.A. (1993). *The origins of order*, Oxford, UK: Oxford University Press.

[24] Kessler, J. (2013), How many pages of regulations for 'Obamacare'? Washington Post, 5 May 2013, Retrieved from: http://www.washingtonpost.com/blogs/fact-checker/post/how-many-pages-of-regulations-for-obamacare/2013/05/14/61eec914-bcf9-11e2-9b09-1638acc3942e_blog.html

[25] Li, M. and Vitányi, P., (1997), An introduction to Kolmogorov complexity and its applications, New York, NY: Springer.

[26] McCabe, T.J. (1976), A complexity measure, *IEEE Transactions on Software Engineering*, SE-2(4), 308-320.

[27] Miller, G. (1967), *The psychology of communication*, New York, NY: Basic Books.

[28] Simon, H.A. (1981), *The sciences of the artificial, second edition*, Cambridge, MA: MIT Press.

[29] Stangler, D. and Arbesman, S. (2012), *What does Fortune 500 turnover mean?* Ewing Marion Kauffman Foundation.

[30] Taleb, N.N. (2007), *The black swan*, New York, NY: Random House.

[31] Taleb, N.N. (2012), *Antifragile: Things that gain from disorder*, New York, NY: Random House.

[32] Willingham, D.T. (2009). *Why don't students like school?* San Francisco, CA: Jossey Bass.

[33] Watts, D.J. (2003). *Six degrees: The science of a connected age.* New York, NY: Norton.

[34] Wood, R. (1986). Task complexity: Definition of the construct. *Organizational Behavior and Human Decision Processes*. *37*. 60-82.

[35] Zuboff, S. (1982), New worlds of computer-mediated work. *Harvard Business Review*, 60(5), 142-152.