A Field Guide to Educational Simulations

By Clark Aldrich

Published by ASTD

Clark Aldrich is involved in several long-term e-learning research projects. In one, Aldrich is building a model of a high-end immersive simulation that can be found at www.simulearn.net. He also speaks to, consults with, and writes for implementing enterprises, suppliers, publications, and venture capitalists. He can be reached at clark.aldrich@att.net.

ASTD's *Field Guide to Educational Simulations* is part of a *Learning Circuits* series; each guide is a concisely written, practical guidebook that provides in-depth coverage of a single topic vital to e-learning.

Linking People, Learning & Performance

The formal education business has a huge problem.

The next generation of learners, roughly those age thirty and younger, have grown up playing computer games. These once and future learners have learned how to learn through interactions with computers. They expect to be engaged on multiple levels simultaneously, in a fast-feedback, graphical, high stimulation, extremely immersive, user-centric environment.

As a result, they're utterly bored in traditional classrooms. Their ability to process lectures that last more than 30 minutes is suspect. Indeed, Harvard professors are complaining that law students begin to fidget after 45 minutes of lecture—or worse, they start playing solitaire on their laptops.

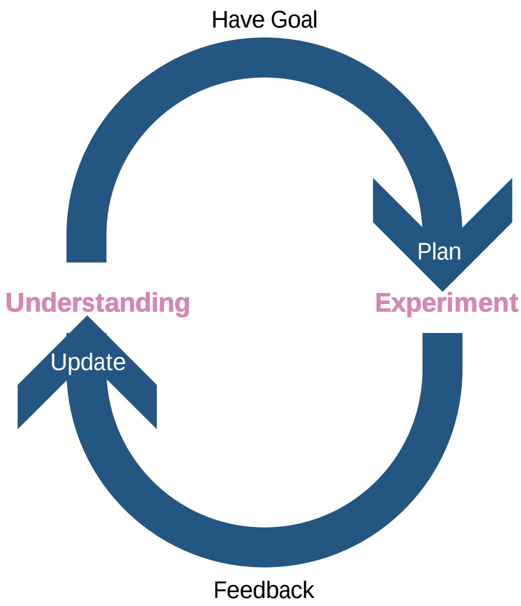
Many instructors and trainers are wondering to what degree computer games can be part of a solution to this problem. Can the lessons, techniques, and technologies of computer games be intelligently applied to create a new breed of formal learning simulations? For instance, US$20 million flight simulators already exist to instruct military and commercial pilots. Can that model be extended to desktop computers for new audiences?

Perhaps the question should be, Can training programs aimed at Generation X and beyond succeed if they ignore simulations? The answer: not very likely. Therefore, simulations are sparking excitement in the e-learning world. And yet, simulations—especially soft skill simulations—are approaching the peak of inflated expectations. Everyone's talking about simulations, but few have seen a model they like.

This document will not attempt to make the compelling case for simulations. Nor will it evaluate all of the simulation-based learning programs currently on the market. Instead, in the spirit of teaching people to fish, this document will present a framework for looking at and understanding simulations to help you become an effective consumer, deployer, user, or producer.

Full-cycle learning

The premise of simulations is simple: Learning is only effective full-cycle. Learning starts at an initial understanding, moves to testing that knowledge, and ends at building a more refined understanding.



We can parse the cycle into the following steps:

Step 1: Understand a system. This understanding may be incomplete to varying degrees. The broadest understanding could be summed up in the German word *weltanschauung,* which roughly translates to mean "a view of the world." For example, one understanding might be a vague memory that a hotel is about 10 miles from the airport, just off the main highway, with an exit ramp after a movie theater.

Step 2: Have a goal. The goal might be driving a car from the airport to the hotel. A larger goal can be broken into smaller sub-goals, such as finding the exit number to the hotel. People use their understanding to formulate a plan that they then execute.

Step 3: Receive feedback. Feedback isn't necessarily either good or bad. In the hotel example, one might take a wrong turn, but in doing so, stumble across a good restaurant or the location of the next day's meeting. The lesson learned isn't just to plan better. The trick in some types of simulations is to make mistakes as educational as successes and allow for serendipitous learning.

Step 4: Update knowledge. Regardless of the degree of success, new information or outright failure creates feedback that updates the person's understanding of the original goal.

Then, the cycle begins again. Many week-long classroom experiences focus three or four days on building a shared understanding. By Thursday, the learners participate in some role-playing activities that challenge their understanding. The goal of all e-learning programs, and the real implication of blended learning, is to give smaller, more frequent pieces of new information that can be tested, accepted, rejected, and honed immediately.

Six Types of E-Learning Content

When talking about e-learning, it's useful to set the stage. There are six different types of e-learning content which need, in their own right, to be procured, created, valued, and managed differently. The following two models are dominant today:

Extended books. Most asynchronous courses are basically workbooks delivered over the Web. The material is trustworthy and up-to-date, at least to the same standard as a magazine.

Extended lectures. Also called virtual classrooms or synchronous and live e-learning, extended lectures are modeled after the traditional lecture model, but enable teachers and learners to connect from different locations

In addition to simulations, there are three new types of content emerging that will ultimately dwarf the use of traditional e-learning types.

Extended community. People are learning from organized and ad hoc communities, such as Internet chat rooms, blogs, and some knowledge management implementations. The material is often very current, but it's the user's task to separate among information, misinformation, and disinformation.

Extended access to experts. Soon, it will be easier to find the person—internal or external—who has the answers to your problems, and get them to help you. From a practical perspective, today's call center workers and IT help desk staff are early pioneers.

Embedded help. The vision of embedded help is to give a user just enough help on how to use the program exactly when they need it. Today, an example of embedded help is an online dictionary. In embedded help, part of the system figures out when a user seems lost and tries to suggest a solution.

All of the new content types are going to cause angst for the traditional training organization. In many cases, they will be hard to track, inconsistent, hard to charge-back, and increasingly blended into real activities rather than sequestered. Most important, however, all six types of content will co-exist. The goal isn't to create any implied hierarchy but to learn how enterprises and individuals can best use each content type.

Enter computer-based simulations

The best place to start any discussion of simulations is with a quote from Will Thalheimer in *Work-Learning Research,* "There are two types of simulations: Those that actually simulate something real and those that are called simulations by their developers but don't actually simulate anything."

Currently, the marketing value of the word simulation is high, but only a small fraction of the suppliers that claim to provide simulations actually do so. Having said that, ultimately, the goal of this paper isn't to determine what is or isn't a simulation (or, in some cases, a simulation of a simulation). Instead, it's more important to determine

❍ different simulation models
❍ pros and cons of each model
❍ criteria to evaluate each model
❍ when specific models should be used.

The right simulation

The first decision around creating, buying, or customizing a simulation is determining the subject area that it will cover. Famous trial lawyer Garry Spence has often said that the best way to win an argument is to be on the right side. Perhaps the simulation corollary to Garry Spence might be, the only way to create the right simulation is to simulate the right thing.

Historically, simulations have been successfully used to accelerate learning of products, such as Microsoft Office or a new camera. They've also been effective at explaining a new situation, such as financial planning or a military invasion. Perhaps the most effective simulations have involved both device and event, in which the interface is the control, such as a flight simulator for an emergency landing or nuclear power controls during a crisis.

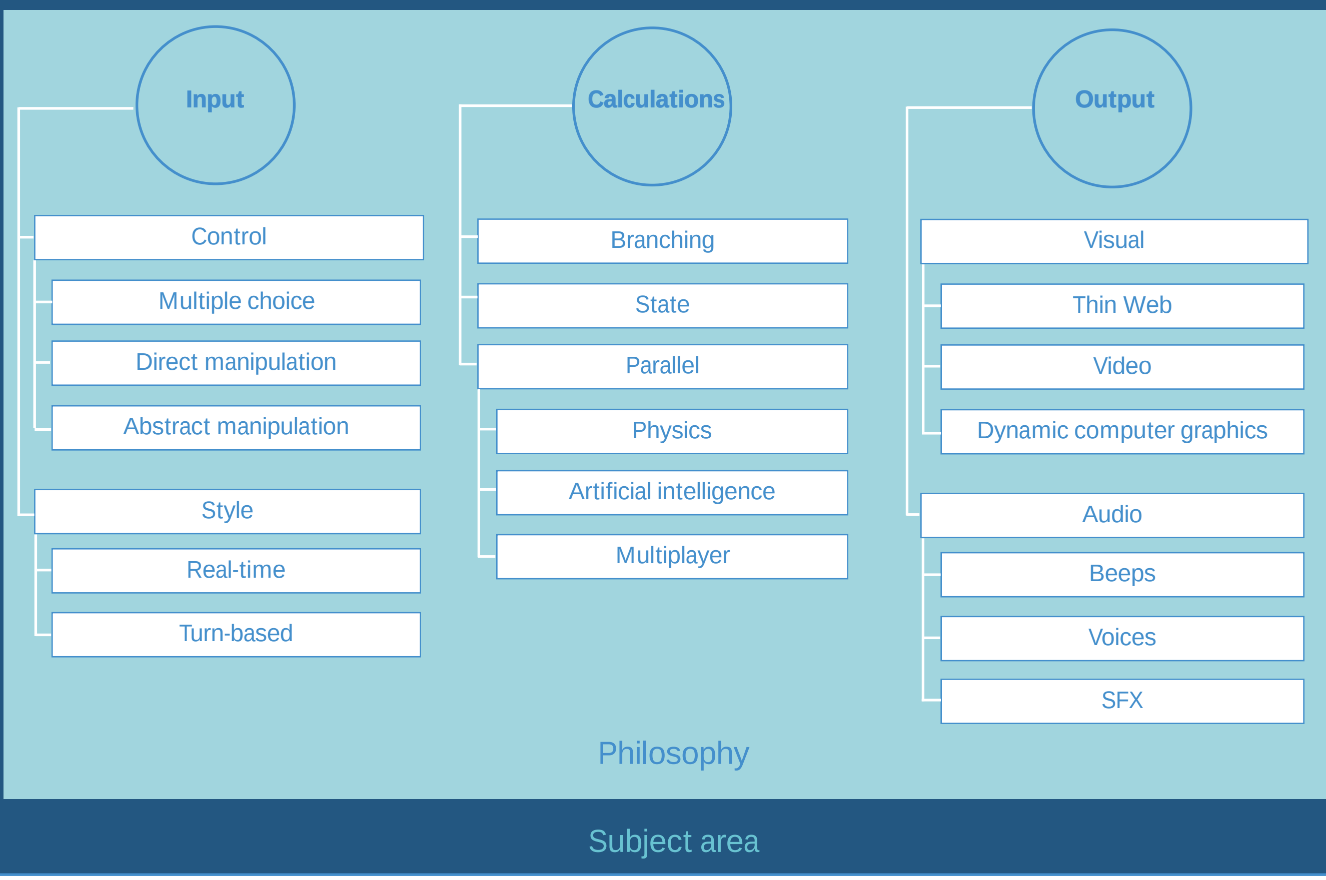
Jane Boston of LucasLearning broadened the scope of simulations by saying, "Simulations are best used in three ways. First, they're ideal for developing an understanding of big ideas and concepts—those things for which experience alone can deepen understanding. As such, simulations are great for dealing with time and scale. Second, simulations are good for situations where it's important to give people practice in decision-making before they're faced with a dangerous or critical real-life situation, such as those used for emergency personnel. Third, simulations are wonderful resources for taking us to a time or place that we're unable or unlikely to experience directly."

Choosing the philosophy of the simulation may be the single hardest task. In most cases, this is where people have to decide what not to simulate as much as what to simulate. Simulating everything is a sure route to failure. For trainers, this is where old knowledge breaks down and instructional design methodology no longer applies. The need for performance objectives actually grows, but the number of ways to complete them becomes overwhelming.

Building a simulation

The workings of a simulation are similar to the basic model of the learning activity that it represents. Users express themselves through their input, the simulation performs calculations and, in turn, presents feedback in the form of an output. Then the process begins again and continues until specific conditions are met.

Figure 1: Simulation Components



Input: A key decision to determining input is the simulation's interface. *SimCity* and *The Sims* creator Will Wright recommends a metaphor approach. "People bought SimCity thinking it was like a train set, and the interface reinforced that. [The metaphor] made it accessible. But they came to realize the game was in fact more like gardening. Things sprout up—you have to plant and you have to weed. Players have to update their mental models to be successful." As the general interface is chosen, there are different forms of control and various engagement styles to consider.

Multiple choice input

One of the most familiar forms of simulation input is multiple choice. Generally, users are presented with two to five options, and they pick one from the list. Variations of this model include filling in forms with either words or numbers.

Example from *QuickCompliance*

Table 1: Multiple-Choice Input

Pros	Cons	Good for	Criteria
❍ Simple to figure out ❍ Cheap to construct ❍ Gives the user new information ❍ Options can foster contemplation of new possibilities	❍ Can over-guide the user ❍ May railroad people into a decision that they're not committed to. "I don't want to say A or B!"	❍ Courses that require minimum preparation time for the user ❍ Entry-level employees	❍ Does it have sufficient richness of options? ❍ Does it capture the right choices? ❍ Is it too easy?

Direct-manipulation input

When simulating a product, a very effective input mechanism is to replicate its characteristics, such as buttons and switches. Probably the best known example of a direct manipulation interface is Microsoft's Solitaire, in which the e-cards are moved in the same manner as real cards.

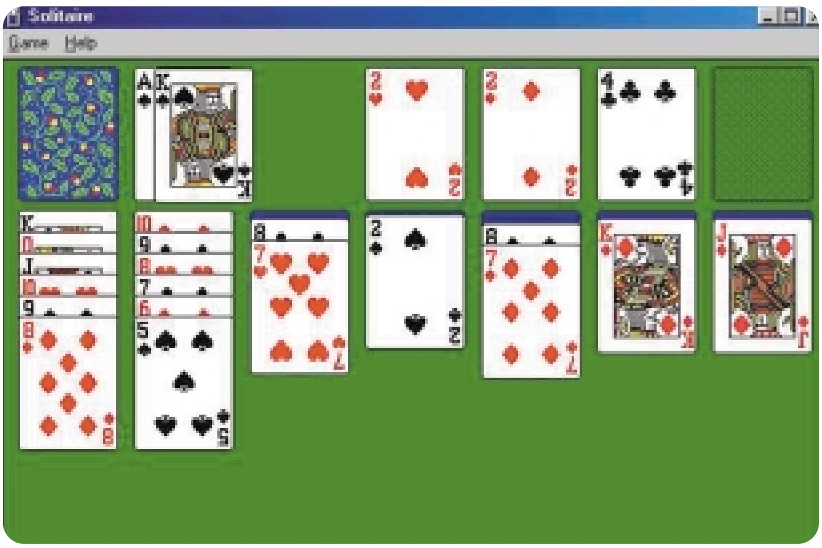


Table 2: Direct-Manipulation Input

Pros	Cons	Good for	Criteria
❍ Feels real	❍ Mildly expensive to create	❍ Some product demonstrations	❍ Does it provide an accurate representation?
❍ Action is natural	❍ Sometimes confusing	❍ Kinesthetic learners	❍ Are users able to backtrack if they make a wrong move?
❍ Can be achieved using a browser		❍ Mechanical skills	

Abstract-manipulation input

The interface for a product simulator is almost necessarily the product itself. But for soft skill simulations—and even the more open-ended parts of a product simulation, such as contacting air traffic control in a flight simulator—more creative approaches are necessary.

This often involves complicated indirect or abstract manipulation of on-screen icons. Increasingly, the keyboard or mouse controls the objects' actions directly, without any on-screen indicators.

Example from *The Sims*

Table 3: Abstract Manipulation Input

Pros	Cons	Good for	Criteria
❍ More options	❍ Very expensive	❍ Technically sophisticated audiences	❍ Is it intuitive?
❍ Interface can introduce a way of organizing disparate information	❍ It's a commitment; needs instructions just to use the simulation	❍ Committed audiences who can ramp up the learning curve	❍ Is it responsive in real time?
	❍ It's as much art as science	❍ Generation X users and younger	❍ Are controls aligned with the task?

Turn-based input

Some turn-based simulations give the user as much time as they need to make decisions. Like chess, users can survey the situation, do research, and experiment with some different possibilities before committing to an action.

Table 4: Turn-Based Input

Pros	Cons	Good for	Criteria
❍ Less expensive ❍ Promotes contemplation, thoughtfulness, and reflection	❍ Not like real life ❍ People learn to manipulate the simulation	❍ Situations that aren't inherently real time, such as financial analysis ❍ Customer-driven applications that need to be highly positive experiences	Is there some sense of flow between various states of action or play?

Real-time input

Other simulations require users to interact in real time. Doing the same thing a minute later changes everything.

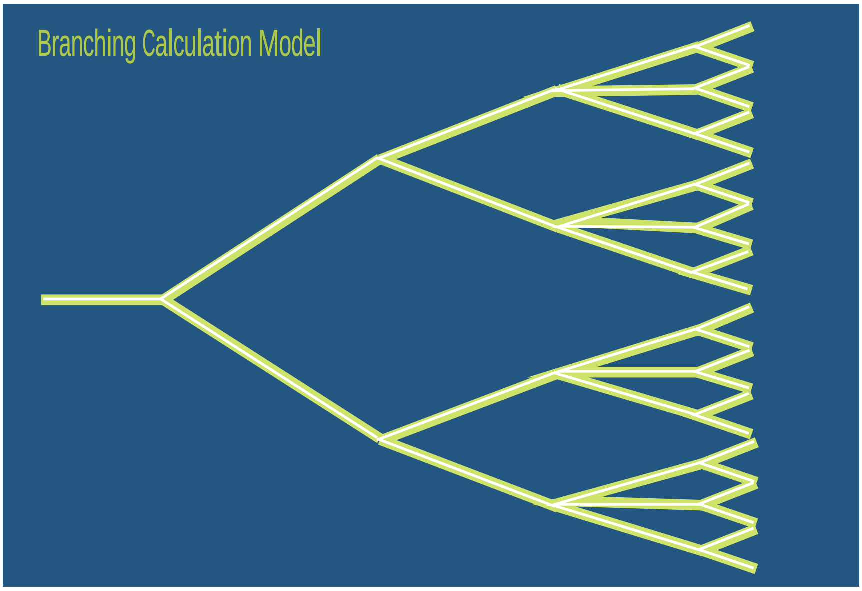
Table 5: Real Time Input

Pros	Cons	Good for	Criteria
❍ Feels more like real life ❍ Action is engaging ❍ Challenges users emotionally through an increase in pressure ❍ Allows the player to see a flow of behavior ❍ Explains the *what* and *when* of an action	❍ Expensive ❍ Can move at a fast pace, creating an unfair advantage to those with faster reflexes	❍ Skills that need to be applied under pressure ❍ Skills that require the understanding of a fluid and continuous system	❍ Will there be a smooth interface? ❍ Will the flow of information be either too slow or too rushed?

Calculations. The heart of the simulation is its calculations. There are three broad categories of simulation calculation styles, each with advantages and disadvantages.

Branching calculations

Early business simulations most commonly used branching schemas. Telephone dialing is a good example of a branching system. Every digit is a branch, the order matters, and at the end you're at one distinct destination.



Branching calculations are an interactive assessment model that's based on early military manuals. In these manuals, engineers would encounter a problem for which they would be given a multiple-choice test. Each answer led the users to another page. On the new page, they would be told whether they were correct and how to proceed, or they would learn that their answer was incorrect and exactly why they were wrong.

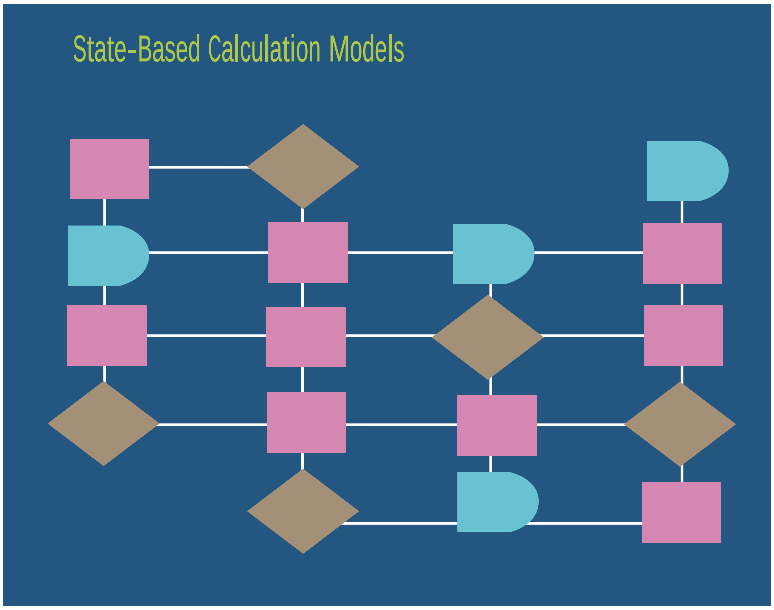
While not technically a simulation, these branching models are at least a simulation of a simulation. They provide a rich opportunity for storytellers to control the situation. Also, these are the simplest calculation models to create.

Table 6: Branching Calculation Model

Pros	Cons	Good for	Criteria
❍ Employs an adaptive assessment model ❍ Can very precisely embed linear instructional content at the right time because you know exactly how someone got there ❍ Cost-effectively simulates a simulation	❍ More of an assessment than experience ❍ Can feel manipulative if players can't do what they want to do ❍ Can feel confining to the user ❍ Can require a lot of hand-holding	❍ Task learning ❍ Consistent assessment ❍ Teaching software skills ❍ People who like telling stories; user/authors have complete control over the experience, from beginning to end ❍ High-level sequencing messages, or when a very specific outcome is needed	❍ How thick are the branches? ❍ How rich is the option set?

State-based calculations

State-based calculations, uses an open-ended exploration model. For users and learners, there's a sense of freedom and some room for play. For example, entering a state-based system is similar to entering a museum. You can start by going to the dinosaur room. Then, you can explore the exhibit on Eskimos. Later, you can visit the room with information about whales. If you want, you can return to the dinosaur exhibit.



To extend the analogy, there could be a guard between two doors that will only let you through if you have a pass. Therefore, you will have to walk to the lobby where you can buy the pass before you can get into the room.

As with branching models, the simulation knows exactly where the user is. Unlike branching models, however, the program doesn't know how he or she got there.

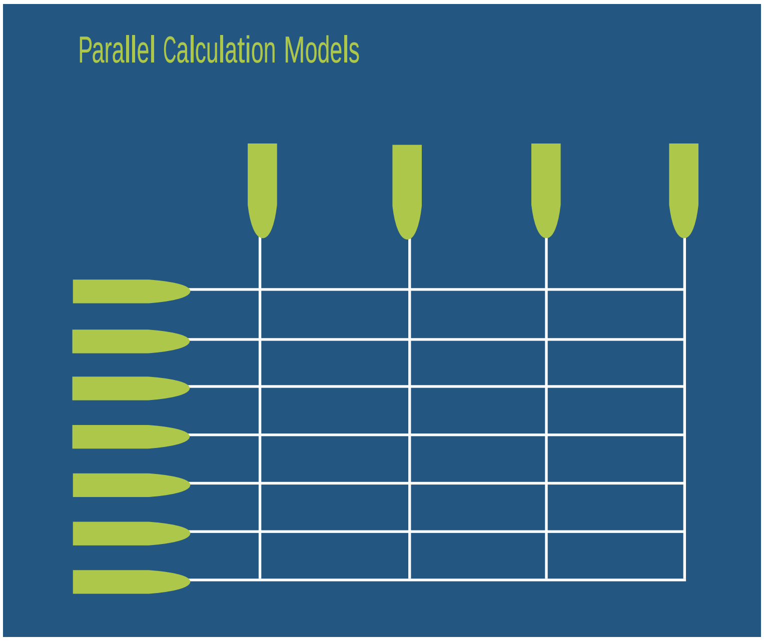
Table 7: State-Based Calculation Model

Pros	Cons	Good for	Criteria
❍ User has a lot of freedom ❍ Straightforward to build ❍ User can back-track during the simulation ❍ User can repeat the experience	❍ Users can get lost ❍ The world is quite static	❍ Conceptual learning ❍ Customer-focused applications	❍ How dense are different action states? ❍ What are the connections between different action states?

Parallel calculations

For better or worse, the parallel model comes closest to representing the actual process of doing something. Consider a simulation of driving a car. You're at a specific location on a map, moving in a particular direction at a certain speed. The world may have other characters, controlled by an artificial intelligence agent, such as aggressive or incompetent drivers. There are general rules for handling a vehicle, and there could be a large number of additional rules playing out almost simultaneously, such as

❍ tires have less traction when it rains, and even less when it snows
❍ the wheels on the car turn in relationship to the steering wheel
❍ the key turns the car on and off
❍ heavier cars decelerate less quickly than lighter cars.



At the low end, the rules for this model could be captured in a medium-sized spreadsheet. At the high end, this model could encompass thousands of rules that influence one another.

Game designer Warren Spector (of *Deus Ex* fame) says, "What you want to do is create a game that's built on a set of consistently applied rules that players can then exploit however they want. Feedback [tracks] player choices so they can make intelligent decisions to move forward based on earlier experience. In other words, rather than crafting single-solution puzzles, create rules that describe how objects interact with one another (for instance, water puts out fire), and turn players loose."

SimCity creator Will Wright also notes, "The more creative the player can be, the more they like the simulation. There's nothing more satisfying than solving a problem in a unique way." He adds, "Another aspect is being able to describe yourself to the game either directly or through your actions, and watching the game build around you."

Even the term parallel can be further defined. There's often a physics system, which describes the rules, and there's an artificial intelligence system that controls any computer-controlled inhabitants.

Multiplayer capabilities in parallel calculations

Within parallel calculations, there may be some multi-player capabilities. The system might involve other real people instead of or in addition to artificial intelligence.

The advantage to a multiplayer approach is that additional people results in complex, adaptive behavior, and the simulation can be increasingly addictive and competitive. The disadvantage is that how the simulation is organized becomes non-trivial, making the overall effect not as realistic as one might hope. Players can realize that there's a system in place and simply try to game or beat it rather than learn. Also, there's significant inconsistency between groups of players, and it's generally more difficult to draw lessons from the experience.

Usually, other characters, whether virtual (AI) or multiplayer-controlled, must follow the rules of the physics system. It's up to the designer, of course, to determine whether the AI component needs to cheat to compensate for less-than-human-like brilliance.

Table 8: Parallel Calculation Model

Pros	Cons	Good for	Criteria
❍ Responsive ❍ Discovery-based ❍ Taps creativity ❍ Learning occurs at different levels ❍ User experience varies	❍ Expensive ❍ User experience varies ❍ Too open-ended ❍ Difficult to mentor users ❍ Needs large amounts of processor power, making it difficult to deliver through a browser	❍ Conceptual learning ❍ Situlations where nuances and small changes can make a big difference ❍ Flight simulators ❍ Military simulators ❍ Computer games	❍ Are rules valid? ❍ What is the number of rules? ❍ What are the game play/ interaction rules?

Output. Clearly, an essential component of a simulation is the feedback it provides in the form of output. There are at least three criteria to look at with any output scheme: ease of deployment, richness of response, and customization or appropriateness of the response.

Light browser-based output

This content can be delivered through the Internet to a standard browser, such as Internet Explorer or Netscape. It can be text, graphs, or small sound and picture files. Increasingly, it could include Macromedia Flash-based animation, Adobe Acrobat PDFs, even small .avi or .mov files.

The amount of output that's possible through a browser-based client is growing, and will eventually encompass all e-learning simulation needs. Unfortunately, current browser-based simulations fall short of some other output models.

Table 9: Light Browser-Based Output

Pros	Cons	Good for	Criteria
❍ Easy to deploy ❍ Plenty of skilled developers in this field	❍ Still needs some plug-ins ❍ Limited density of information	❍ Third-party customers ❍ Organizations with inflexible IT departments	Does browser delivery allow for as much detail and customization as possible?

Video output

This is probably the most familiar media for baby boomers; it's been pushed into the role of educator (with varying degrees of success) since the first documentary.

According to Chip Cleary, president of Cognitive Arts, "We're in love with video. It allows people to observe dynamics of a real world situation, for example, behavior that a clinical psychologist would have to diagnose. But as [our company] builds Web-based systems, we find that we can use still photos and audio with success."

Table 10: Video Output

Pros	Cons	Good for	Criteria
❍ Extraordinary amount of detail and nuance ❍ Feels serious and real ❍ Users age 40 and older have grown up with television and are very comfortable with it ❍ Works well on a low-tech terminal. ❍ Extensive knowledge about shooting and editing video already exists	❍ Expensive ❍ Requires huge bandwidth ❍ User interaction with video has delays ❍ Video clips show trodden ground ❍ The density of information in video also makes it much harder to get right ❍ Can be too real for off-the-shelf, making it harder to extrapolate rules ❍ Making small changes in video is difficult	❍ Custom rather than off-the-shelf solutions ❍ Very specific behavior role-modeling ❍ Situations in which very high density information is critical, such as differentiating between fruit at a supermarket chain	❍ Is on-screen information for a given environment accurate? ❍ What are the cost considerations? ❍ What are bandwidth limitations?

Computer graphics output

There are several benefits to computer graphics. However, while simulation's move to computer graphics may be inevitable, it's been slower than many had hoped.

According to Imparta's CEO Richard Barkey, "Real actors in a sim seem to have the benefit of building credibility and allowing subtle nuances of body language and expression. However, they're limiting, and I expect computer graphics will take over in the next twelve months."

Clark Quinn of OtterSurf Labs adds, "Video is great when contextualization is important, but animation is better when conceptualization is the goal. We're actually well prepared to process comic strips and cartoons, and [animation] can strip away irrelevant details and exaggerate important elements more effectively than video, under particular learning outcomes."

Table 11: Computer Graphics Output

Pros	Cons	Good for	Criteria
❍ Easy to tweak ❍ Taps creativity of user ❍ Can explore uncharted territory; with computer animation, one can actually be somewhere that no one else has ever been ❍ Generalizes skills	❍ Expensive ❍ Skilled developers are difficult to find ❍ Requires significant processor power, graphics cards, DirectX software ❍ Some things that look like dynamic computer graphics are really pre-rendered	❍ High-level business issues ❍ Skills that need to be generalized over different and unpredictable situations	❍ What are the resolution requirements? ❍ What are the smoothness requirements? ❍ Is it appropriate given the available processor power?

Simulation archetypes: There are plenty of combinations and permutations possible but four types of simulations are becoming standard: do-it-yourself, virtual products, traditional business simulations, and the flight simulator model.

Archetype A: Toolkits for do-it-yourself projects

Simulation toolkits are being introduced into the marketplace as a way of providing the educational benefits of simulations at a low cost with customized corporate-specific content. While the proposition sounds good, it could poison the entire simulation well.

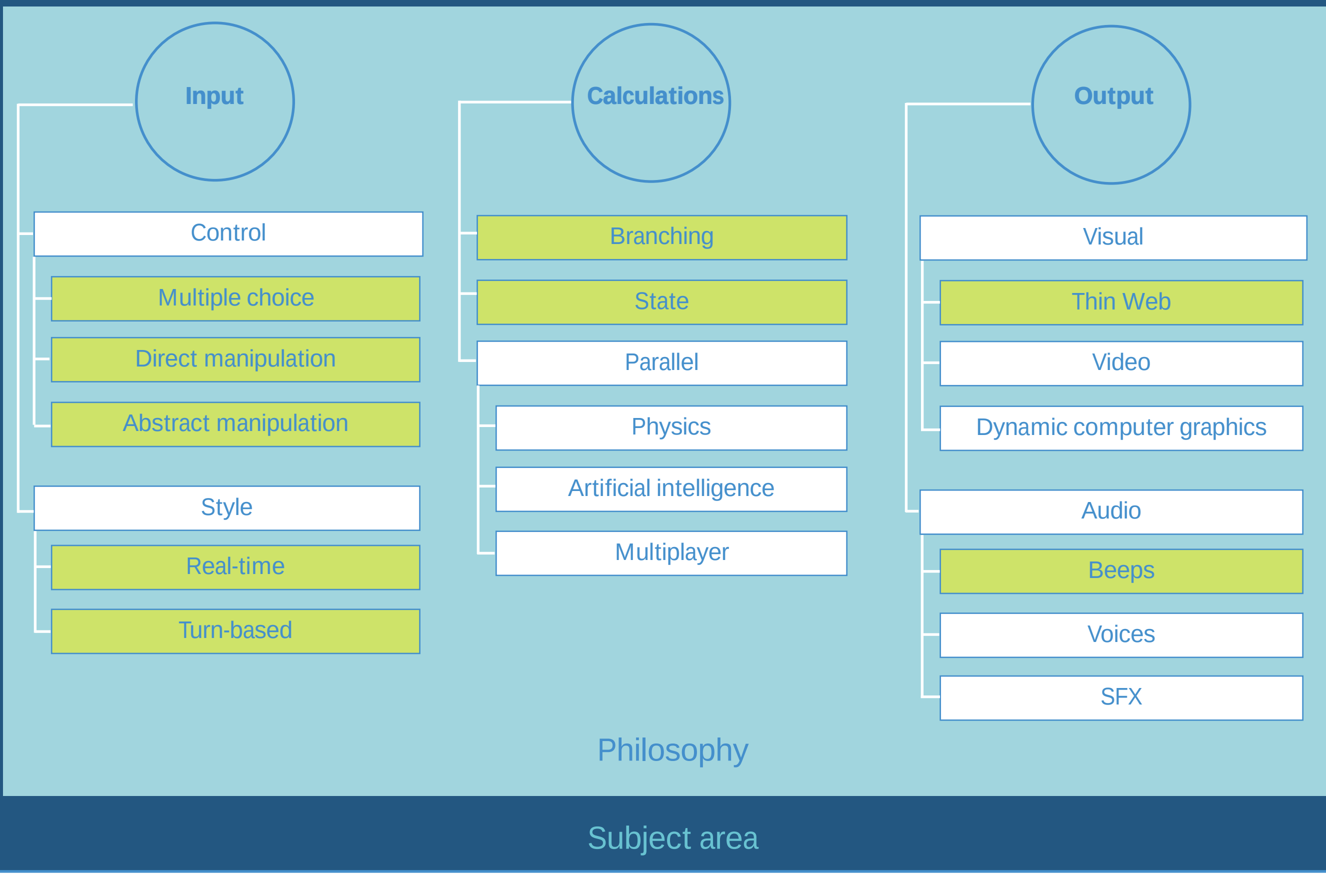
One can't help but make a comparison to the artificial intelligence industry in the early 1980s. They sold a lot of expert systems toolkits, with instructions for enterprises on how to build their own AI systems, but the companies didn't have the competence to build effective systems, and the industry, literally, went away.

For toolkits to succeed, there need to be good professional simulation models. Only then can the approach be generalized. If toolkits proliferate without the requisite skills to use them well, simulations will go away—at least for several years.

Development tools include

- ❍ Macromedia Flash, Director, Shockwave
- ❍ Click2learn's Toolbook II
- ❍ LCMSs, such as Knowledge Mechanics
- ❍ Simulation toolkits, such as RapidCBT (Simulis).

Figure 9: Characteristics of Do-It-Yourself Toolkits



Archetype B: Virtual products

This type of simulation may be the surest bet among those currently available in the marketplace. There are many uses of virtual prototyping, such as the ability to

❍ evaluate design and study user error (for example, recognizing and redesigning to avoid interface problems that lead to incorrect operation)

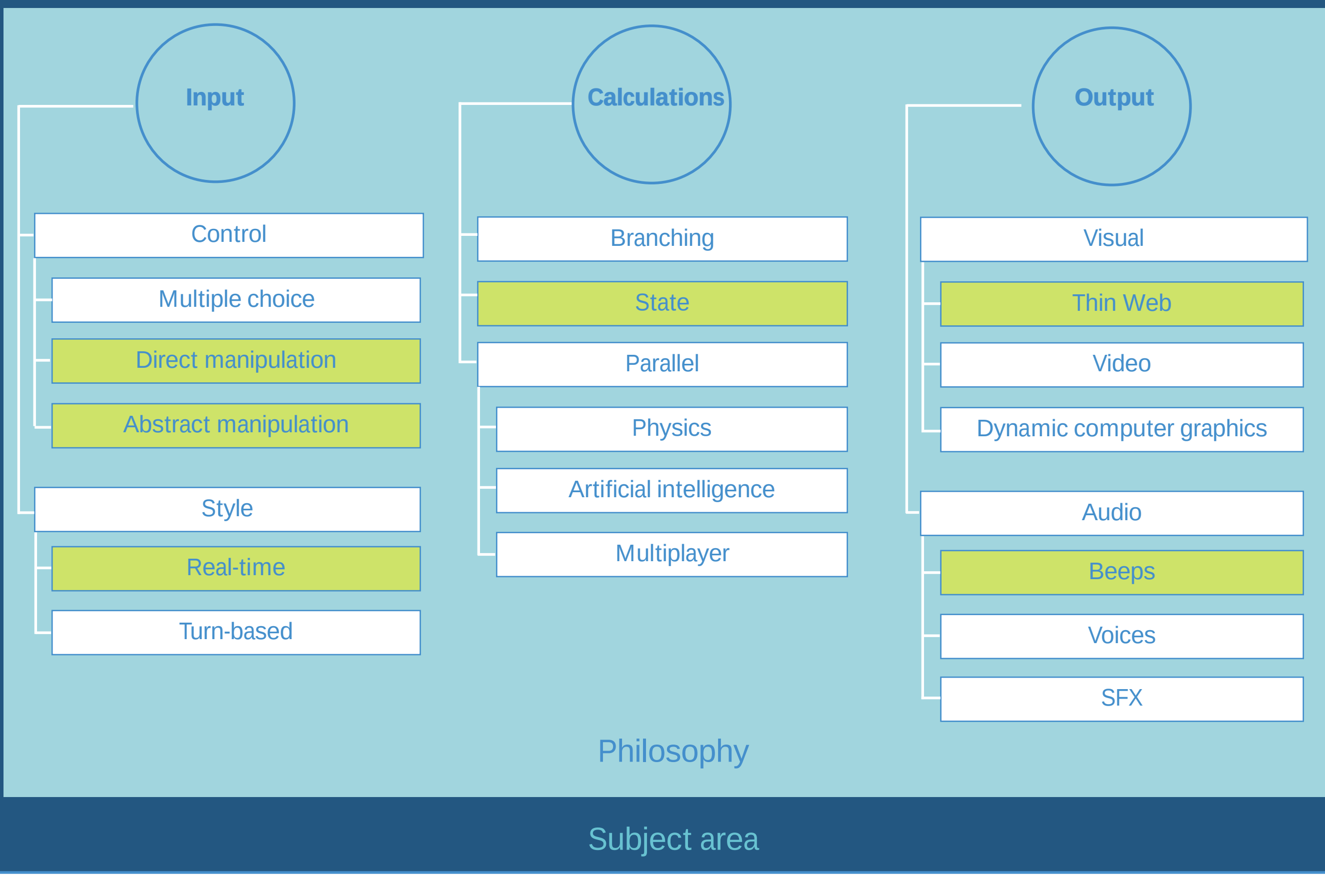
❍ evaluate products (for example, try before you buy)

❍ enable training, build fluency, and assess competency for certifications and high-risk, critical tasks that are difficult or dangerous to reproduce

❍ provide operational support

❍ perform research and experiments to ground theoretical hypotheses or make new discoveries (for example, pharmaceutical research).

Figure 10: Characteristics of Virtual Products



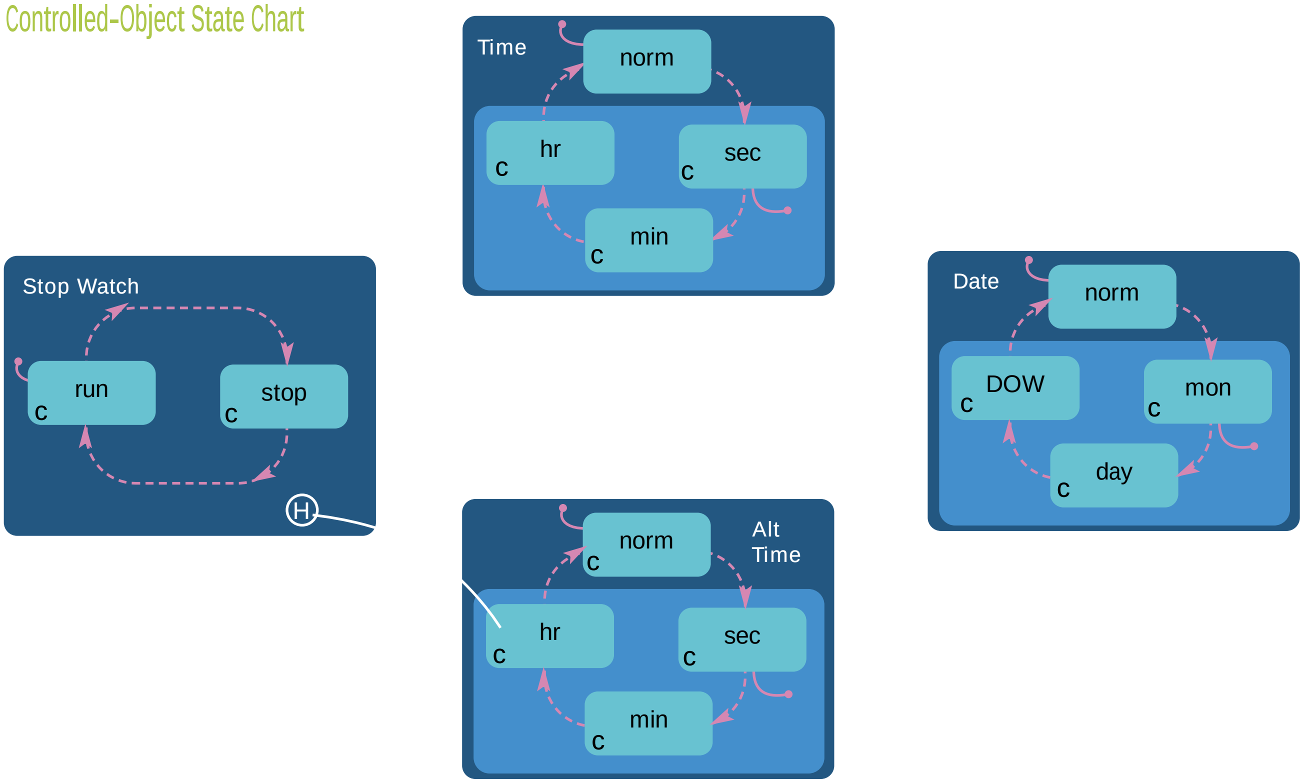
According to the president of Amethyst Research Jonathan Kaye, "Methodologies are being developed that allow people who don't know anything about programming but know how the device works to create charts that are based on how the device and interface function. [They] can hand that to a coder and the process of creating the simulation is mechanical. Subject matter experts can verify [whether] the simulator is behaving correctly without having to know the specifics of the programming, so long as the programmers follow the cookbook."

Not only is the development rapid, but you get an efficient and concise representation that simplifies maintenance and upgrades. Also, you can incrementally develop the simulation, such as by starting from basic behavior and then making the simulation deeper in particular states of action or play.

Development tools include

- MATLAB, LabVIEW, CAD/CAM, ACSL
- development languages and packages such as Fortran, C/C++, Basic, Java, Macromedia Flash, Director, and Shockwave
- Click2learn's Toolbook II
- LCMSs, such as Knowledge Mechanics
- Simulation toolkits, such as RapidCBT (Simulis).

Controlled-Object State Chart



Archetype C: Traditional business simulation models

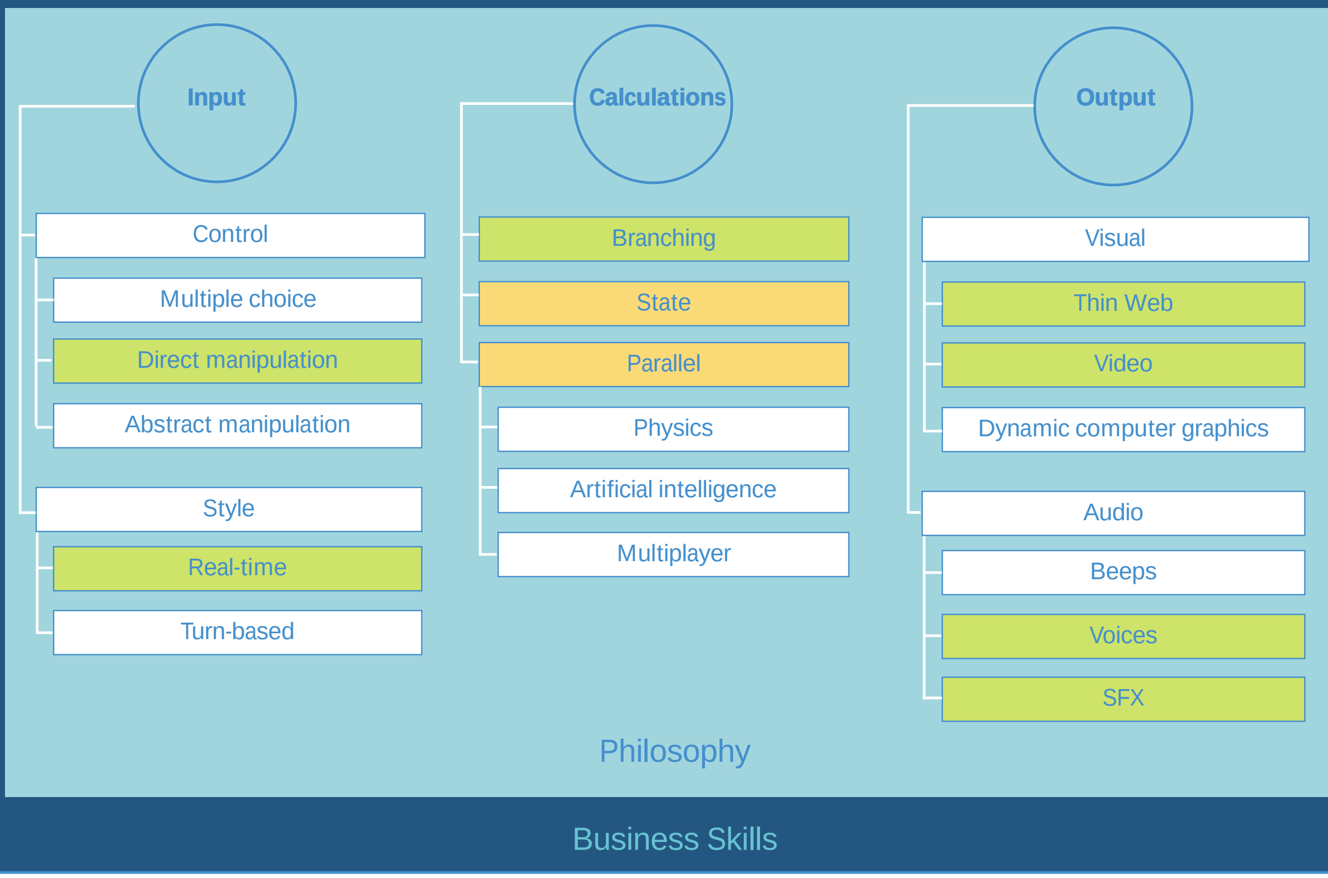
Many early business and several current low-end simulations use a branching mechanism, with static content at the nodes. Even if these simulations are ineffective, it's important not to discount the technique.

This model of branching media may be perfect for new entry-level employee training that needs to convey very specific information, such as how to park a car for a rental car company, how to greet a guest at a hotel, or how to set a table at a restaurant.

Suppliers include

- Ninth House
- Indeliq
- Cognitive Arts
- WILL Interactive.

Figure 11: Characteristics of Traditional Business Model Simulations



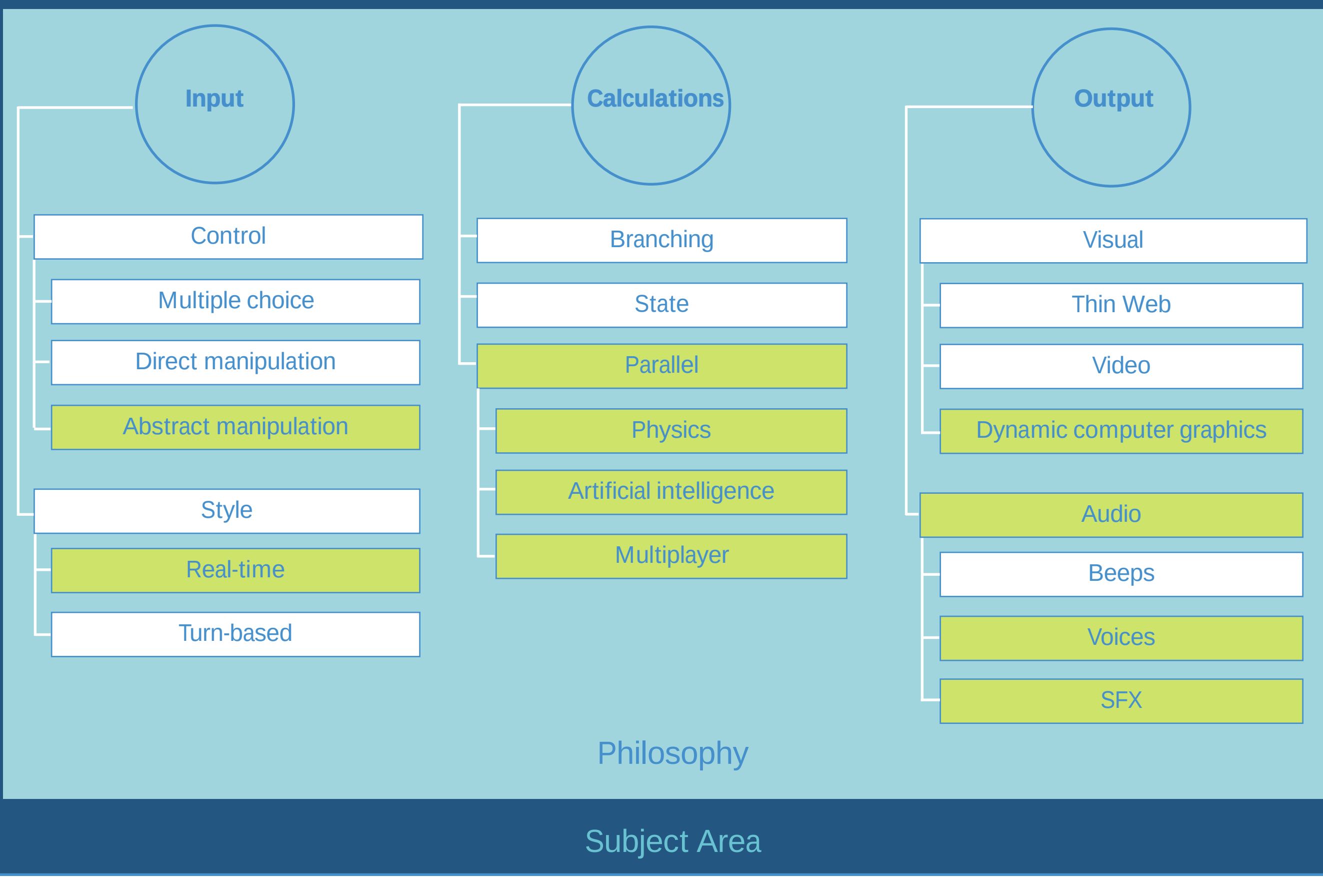
Archetype D: Flight simulator model

Computer games and military sims use the flight simulator model, which allows for a myriad of states and transitions between states. These simulations often create the reaction of, "I know I've learned something, but I'm not sure what it is."

Currently, it takes longer than a year to develop a good flight-type simulation. This process can't be rushed with more people. To quote the old criticism of IBM, you can't have nine women each be one month pregnant. More important, these sims can easily cost more than US$1 million to create. As a result, there won't be that many around until at least 2003.

Indeed, with few prototypes available, the training industry is working hard to categorize the flight simulator model into something familiar to them: "These are like avatars on the Web," or "This is like *The Sims*." The comparisons to familiar experiences are more often wrong than right, and they may lead decision makers to a faulty understanding.

Figure 12: Characteristics of Flight Simulator Model Simulations



Do Brand Names and Gurus Help or Hurt?

It has been assumed, especially by me, that using a brand name subject matter expert (SME) helps a simulation. The brand name SME could be a person who has written one or more successful books in an area, a well-known successful businessperson, a famous speaker/pundit, or a dean-organized team from an Ivy League business school program.

From the persepctive of a simulation content creator, the expectation is that the SME adds value at several points:

- They provide a lot of content.
- They're expected to shape a better experience.
- Their faces adorn the Websites, adding percieved credibility.
- They can push products to their clients.

From a procurer's perspective, the assumption is that SMEs add credibility to the content and eases the marketing of the content to the corporation's learners. Those assumptions may be false. In fact, the opposite may occur.

For a content creator, the simple act of procuring a brand name can take months. Further, a SME's involvement can easily cost--in both time and money--15 percent or more of the entire project's budget. Again, that's money and time taken from development.

The content that gurus and brand names provide tends to be linear, such as notes from upcoming books, speeches, or war stories. But simulations require a depth that few broad specialists can touch. Asking them specifics is a careful art. They can add months to development time as reading over any documents gets pushed back for other engagements. Any changes they do make tend to be ego-centric.

Learners, meanwhile, care more about the experience than the name. Taking a simulation endorsed by a brand name is like eating a box of Wheaties with a famous athlete on the box. The cereal may be good, but the expert is just veneer.

Imparta's Richard Barkey accurately summed it up this way: "The role of the subject matter expert is to write the simulation. It's the only way you get the depth and subtlety in the environment. And I prefer practitioners to purely academic SMEs. You can't create anything useful from a PowerPoint pack and three days snatched from busy b-school politics."

Simulation issues

There are and will continue to be issues around simulations for some time.

Cultural shift. Simulations and other types of new content are going to be deployed, taken, used, and measured differently. As do other new content types, simulations contradict the e-learning trend towards increasingly smaller slices of content. Instead, they present larger experiences that require contemplation rather than simple answers.

Evaluation. It's difficult to evaluate simulation-based content, which is only part of the process and is distributed widely throughout the experience. At this point, there are no credible instructional design theories for evaluating simulations.

Free play versus guided interaction. One of the most significant long-term issues will be, how much do you guide the user? How rich should the instructions be? Too much assistance dilutes the impact and learning value of the simulation—users who receive linear instruction may only be mindlessly carrying out orders. If there's too little guidance, users may get lost and misunderstand what they learned or what the goals were.

Identifying appropriate subject matter experts and designers. If today's subject matter experts and e-learning designers are inadequate, where do we find the next generation? Simulations require a new way of thinking about content that may never truly be understood by the current generation of developers.

Gaming simulations. One fear about simulations is that people will learn how to beat them rather than learn from them. One real example comes from the computer gaming industry's highly successful *Roller Coaster Tycoon.* In the game, users must successfully build their own amusement park and be evaluated on customer satisfaction. Here's an actual hint from a Website called gamewinners.com on how to increase your park rating: "Drown all of the unhappy or angry guests. Eventually, your park rating will go up 100 to 200 points." Clearly, this is not a technique suggested by Disney.

CD-ROM/hard drive-delivered versus Web-delivered. Is it better to use hard disks and processor power or a lot of bandwidth? Simulations that require hard disk space and high processor power take more work to install but have a low impact on the long-term IT infrastructure. Simulations that require high bandwidth are Web delivered, which is easier to deploy up front but requires huge network bandwidth during use. Multi-player games, especially those employing rich output models, still require significant bandwidth, even if they're installed on the client.

Toolkits versus finished products. When the artificial intelligence boom of the early 1980s was in full swing, vendors were selling AI toolkits to large enterprises to build their own expert systems. Few enterprises did so successfully, and all of the vendors disappeared. Toolkits will eventually be popular, but not until successful simulation models have propagated the market.

Do you know what you know? It's harder to test simulation-learned material than textbook data in traditional tests. As with life, people might learn different things.

Role of standards. Right now, standards are far from being able to handle any of the emerging content types, including simulations. Likewise, LMSs have a difficult time tracking them.

Looking ahead

Perhaps the best conclusion to the simulations conversation is to think of the DOS/Windows analogy. There was a time when most computer programs used the DOS operating system (for you computer people, yes, that is redundant). Then, Microsoft's Windows came along.

At first, this graphic user interface was good only for a few areas. It co-existed with DOS. The IT departments fought it. They said it was too game-like and costly. It required hardware upgrades for the oldest computers. New applications were also more expensive. Still, early DOS programs were ported into the Windows environment, a forced, awkward fit that had IT people crowing, "See? I told you that this is no better."

But Windows applications were more accessible to many more people. Windows was user- and business-centric, not IT-centric. People who had been experts in DOS computing were increasingly obsolete.

The number of applications grew. New users were passionate about Windows and fought for it. Eventually, although it took a lot longer than it seems looking back, Windows applications won. Windows still co-exists with other operating systems, and it's not good for everything. But computing never turned back from the graphic user interface.

To fully appreciate simulations, you have to look at a culture that fully appreciates simulations, such as the military. Says Ed Glabus of MindSim (recently acquired by Aegis Research Corporation), "Those mid-level managers use simulations the way that corporations use spreadsheets. They wouldn't think of doing an operation without simulating it first."

A final thought

Most media, including magazines, newspapers, books, television, and movies have complex ecosystems. There are plenty of niche examples, but there are also some very broad genre-definers at the top, such as *Newsweek* for magazines, *Seinfeld* for television, or *Star Wars* for movies. Another example is the book *In Search of Excellence,* which despite recent admissions, it's fair to say that it launched the successful business book genre by becoming the first must-read. Those examples are the center of cafeteria conversations, meeting references, and email jokes. Those common touchstones define their industries, pulling new people into the industry and reminding people to come back. For e-learning to grow to the next level, it will need standard bearers—great examples that are applicable to a broad audience and define and attract users.

With the Paul Saffo caveat, "A clear view doesn't equal a short distance," I would consider high-end simulations to be the front-runner candidate for creating broad, sharable, e-learning experiences. Until that happens, e-learning will remain a niche industry. But once that happens, e-learning will change the world.

Simulation Organizations

These organizations are examining the role of simulations in education:

ACM Special Interest Group on Simulation
www.acm.org/sigsim

Association of Business Simulation and Experiential Learning
www.towson.edu/~absel

Federation of European Simulation Societies
iatms13.iatm.tuwien.ac.at/eurosim

Games-to-Teach Project, a partnership between Microsoft and MIT
cms.mit.edu/games/education

International Simulation and Gaming Association
isaga.pm.it-chiba.ac.jp

Japan Association of Simulation and Gaming
jasag.bcasj.or.jp

North American Simulation and Gaming Association/Electronic Simulation and Gaming Association
www.nasaga.org

Operational Research Society
www.orsoc.org.uk

PLAY Research Studio
www.viktoria.informatics.gu.se/groups/play

Simulation & Gaming: An Interdisciplinary Journal of Theory, Practice and Research
www.unice.fr/sg

Società Italiana dei Giochi di Simulazione
www.iuav.unive.it/~sigis

Society for Computer Simulation
www.scs.org

Society for Intercultural Education, Training and Research - USA
www.sietarusa.org

The Society for the Advancement of Games and Simulations in Education and Training
graph.ms.ic.ac.uk/sagset

USC Information Sciences Institute
www.isi.edu

USC Institute for Creative Technologies
www.ict.usc.edu