The State of Authoring Tools - Where We've Been, Where We're Going (Jul 15)

by Joe Ganci

July 28, 2015

“Are you happy with the state of tools today or do you find yourself wishing you could do more? eLearning is going to be different for accountants and for surgeons. That means the interactions you build should be different. If you ask yourself, ‘What will my tool allow me to do for this audience and this content?’ then you’re asking the wrong question. The real question should be, ‘What is the best approach to have this audience learn and so what interactions should I build?’”

I just hit my five-year anniversary since I started writing monthly reviews of authoring tools for *Learning Solutions Magazine*. I have appreciated you readers and your feedback very much. During this time, I’ve seen some patterns emerging, both good and not so good, and it’s time for us to step back, take a breath, and discuss what has happened, what is occurring now, and what the future may bring. Not all of this will be pretty, but bear with me to the end. After that, I hope that you will put your thoughts in the comments section below.

Where it started

Many don’t know that eLearning tools have been around for quite some time. In fact, the National Science Foundation funded two authoring tools way back in the 1960s. The first was Plato in 1960 and the second was TICCIT in 1967 (Figure 1). I entered the eLearning world in 1983, while in college, and used TICCIT to create a large Italian course for the university, a project that spanned three years. By that time, both Plato and TICCIT were in full swing and they both had already spawned quite a few other authoring tools.

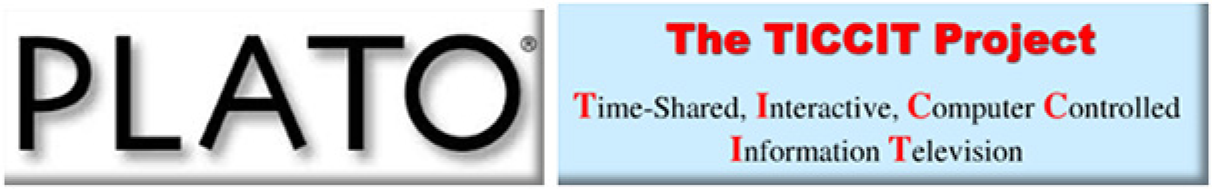

Figure 1: **The first two authoring tools arrived 50 years ago**

Remember that the personal computer revolution started in the late 1970s and of course, over time, PCs have become extraordinarily more powerful. That means that the early authoring tools did not run on personal computers. Rather, they required at least a minicomputer in an air-conditioned computer lab with washing-machine-sized free-standing hard drives that contained a tiny fraction of the memory that your cell phone does now. Learners needed to use *dumb* terminals, meaning that all of them were attached to the minicomputer back in the lab and that all the power was coming from the minicomputer. Some specialized terminals, such as PLATO terminals, actually connected to mainframes.

The tools we used were not simple: they required some programming talent. Hence, eLearning, which was called computer-based training (CBT) or computer-based education (CBE) at the time, meant that we had teams of *instructional designers (IDs)*, those folks who understood learning theory and principles and knew how to design lessons that learners could experience on a computer, and *programmers*, also called *developers*, who would take the storyboards created by the instructional designers, at first typically drawn by hand on paper and later in Word or PowerPoint, and program them on the computer. I found it interesting that instructional designers tended to be more creative, while the programmers tended to be more logical. The two disciplines, when combined well, would result in excellent eLearning.

Yes, that’s right. Even back in the 1970s and 1980s, there was some really good eLearning being created. There were also problems, of course. Instructional designers (nonprogrammers) could not always understand why developers would balk at an interaction they designed. Developers (non-instructional designers) could not always see the importance of the design they were given and would try to change interactions to something simpler for them to program. Arguments ensued, blood was shed, feuds began, wars started.

What happened next

As personal computers became more powerful, eLearning could be created at one’s own desk. I started to notice a pattern at that point and spoke of it in a keynote speech I gave in 2000, both in the United States and in the Netherlands. I called it *Two Steps Forward*, *One Step Back* and I described the phenomenon that was occurring wherein each time we leapt ahead with our technology, we had to make some sacrifices as well, at least at first.

For instance, we were happy when we could stop using floppy disks to store our eLearning because we were given voluminous five-megabyte hard drives in our computers. Imagine, five million bytes! That was great, only we found that in those days before ubiquitous networks it became difficult to move our files from one computer to another. Those were the days when you could often see me take my huge desktop computer and big CRT monitor, with keyboard, mouse, and cables, home from the office some Friday nights so I could continue working over the weekend (Figure 2).

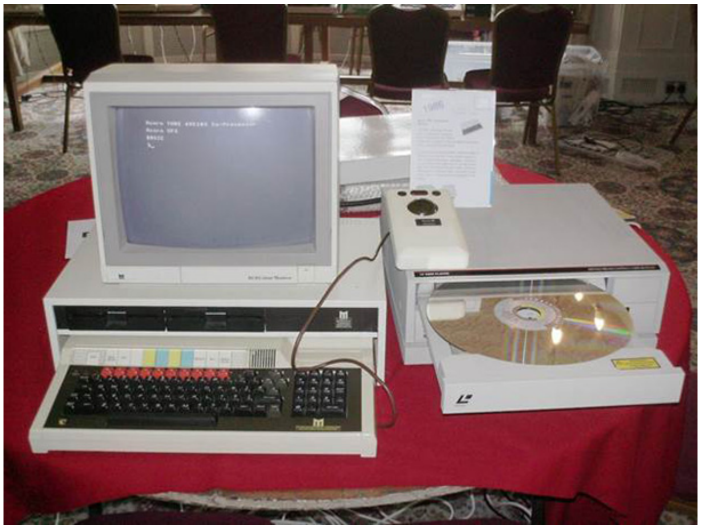

Figure 2: **eLearning, circa 1986: PC and laser disc player**

We also had great big laser disc players that gave us beautiful full-screen video in our learning, but then the digital video files QuickTime and AVI meant we could do away with those expensive disc players. However, in those early days we were lucky if the digital videos could be larger than a postage stamp.

This progress will continue into the foreseeable future. Most recently, we have seen it in being able to deliver our learning to mobile devices. Two steps forward means that we now can let learners access their learning almost anywhere at any time. One step back, though, because we have to deal with smaller screens.

Tools started to change

Tool vendors saw an opportunity to make their tools simpler to use and started to promote them as being so easy to use that an instructional designer could use them. Developers were no longer needed. Many of these tools were add-ins to PowerPoint, which made sense as by 1991 a lot of storyboarding was already being done in PowerPoint. How cool was it that suddenly, from within PowerPoint itself, you could generate eLearning without having to take that second, expensive step to have a programmer make it work? Only in this case many instructional designers found themselves overwhelmed nonetheless and eLearning lessons started becoming more and more just warmed-over PowerPoint files. Many of those tools disappeared quickly, others stuck around.

When dedicated instructional designers started realizing that they were not putting out the best work, they would work with programmers still, at least to do the “hard parts” of the learning, and those programmers often used Macromedia (later Adobe) Flash to create those parts.

Of course, some tools were still powerful enough to use to create great eLearning (based on an ID’s design). One very popular tool was *Macromedia Authorware*, the brainchild of Dr. Michael Allen, who wanted to develop a visual version of Plato. (Figure 3) It became the most popular eLearning tool in history, topping the charts from 1988 until 2005 when Adobe shelved the product after it merged with Macromedia. There are still people using it today, 10 years after it stopped being updated, because nothing like it has emerged. Why? Authorware had become a tool that instructional designers could use to lay out their design and do a lot of the initial work, which they did by dragging icons onto a flow line. Authorware also had its own programming language (and later JavaScript) that would allow programmers to create very efficient and powerful interactions. When Authorware folks got together, they would identify as either *icon-draggers* or *code-heads*—but by and large they were able to work well together.

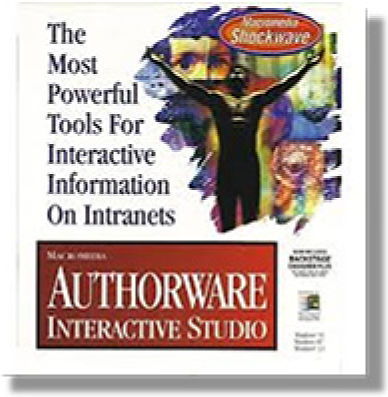
The Most Powerful Tools For Interactive Information On Intranets
Macromedia Shockwave
Macromedia
AUTHORWARE
INTERACTIVE STUDIO

Figure 3: **Macromedia Authorware**

What is occurring now

Remember that initially the tools were meant for programmers, and later on tool vendors simplified them to make them easy enough for instructional designers to use. When that happened, eLearning standards took a dive. We started seeing many more page-turners and PowerPoint lessons with quizzes attached to the end of those lessons. We started seeing eLearning get a bad reputation. Many of us found ourselves hesitant to tell strangers what we did for a living for fear that they would punch us because of the eLearning they had to endure in their companies.

In the last 10 years, we have seen tool vendors change their tools again, not just to meet changes in technology, such as the need to deliver to mobile devices and the impending death of the Flash web player, but also because many have decided to change the audience for their tools once again. While at one time the tools were meant for programmers and later for instructional designers, now tool vendors sought to convince organizations that they could save a lot of money in a different way.

Let me illustrate this by telling you about a recent meeting with a vendor who wanted to introduce me to their new authoring tool. As I sat in their offices, I said, “Before you start the demo, please know that I truly hope that you are not going to say that it’s a tool so easy to use that any subject-matter expert can use it to create great learning.” The person hesitated, took his hand off the mouse, and said, “In fact, that’s what I was going to say.” What followed was a very good discussion about how they wanted to disrupt the market and offer something truly useful and I am now helping them with my suggestions and guidance as to what features they need to include.

So, yes, this is what is happening now. Very often we hear vendors say that we no longer need instructional designers because the tools are so easy to use that Harry the Engineer can create the engineering course himself, or Susan the Physicist can build that physics lesson herself. The bean-counters in those organizations buying those tools are psyched at all the money they can save by not hiring or contracting instructional designers (and of course programmers) to fill their learning needs. They don’t know, of course, that the resulting lessons are often at the very least anemic and at the worst nothing more than boring text and images punctuated with a Jeopardy game and quizzes. Learners end up expecting their eLearning to be onerous and are resigned to getting through it as quickly as possible and in some cases cheating if they can.

So what can be done?

We are seeing a backlash against bad eLearning emerging as people start to realize what has happened. One problem that has held us back is that most of the work we do is performed under nondisclosure, meant to be seen only by a specific audience within an organization. As such, the general public has not seen some of the best eLearning.

But there is hope. As an example, The eLearning Guild holds an event at all their major conferences where attendees can demonstrate their best eLearning examples and everyone votes on the ones that they find most engaging in different categories. Yes, sometimes visuals sway votes more than great coding, but almost everyone recognizes great eLearning interactions when they see them. This leads them to want to improve their own eLearning. The Guild goes so far as to show the winners online, as they will again in August, and each winner demonstrates his or her sample. This allows anyone with access to see great examples that can help improve their own expectations of what good eLearning is all about. If you haven’t seen these, check out the recordings.

There is also hope that some tool vendors are starting to see the wisdom in providing back-end programming capabilities to their tools, so that once again, as in the days of Authorware, subject-matter experts can use the tools to a level, instructional designers can use them at deeper levels, and developers can get down to the deepest levels to make the learning the best possible.

Tool vendors that are successful today do try to balance power and ease-of-use and not just make their tools extremely easy to use. However, no tool today allows for the power that we used to have and that many of us sorely miss. When we use the tools of today, we anguish over the slowness and the inefficiency by which we have to create lessons. The irony is that, in many cases, this slowness means that it is taking longer to create much of the learning today than it did at one time, so organizations are actually paying more. A good programmer can quickly make lessons that work well but that also are easy to maintain and update in the future. However, in many cases those without programming skills are doing the best they can to create lessons that usually have to be thrown out when major updates are needed, leading to even more money being spent.

Last year the Serious eLearning Manifesto was released (see http://elearningmanifesto.org/). Based on the work of Michael Allen, Julie Dirksen, Clark Quinn, and Will Thalheimer, it is a set of 22 principles that should guide any eLearning professional to create the best possible work. I was one of the original trustees, and to date over 800 eLearning professionals have signed the pledge. However, it is difficult to hold to the pledge to create the best eLearning possible when you don't have a tool that gives you the freedom to do so, when you have to compromise your learning design so much that the end result doesn't work well. I believe that the principles in the Serious eLearning Manifesto are wonderful and it is a great opportunity for tool vendors to improve their tools or create new tools that will help us meet those principles. I suspect that those who do will win over the market.

Are you happy with the state of tools today or do you find yourself wishing you could do more? Remember that the best learning possible takes into account the learner audience, the content to be taught, and the context in which the content is to be taught. That means that eLearning is going to be different for accountants and for surgeons. That means the interactions you build should be different. If you ask yourself, "What will my tool allow me to do for this audience and this content?" then you're asking the wrong question. The real question should be, "What is the best approach to have this audience learn and so what interactions should I build?" If you look around and realize that your tool can't handle your needs, find a better one. If you can't find any tool that can truly deliver your vision of great learning, then urge tool vendors to improve their tools further.