Transcript

[0:01] (music)

Derek Bruff: [0:06] This is Leading Lines. I'm Derek Bruff. In this podcast, we explore creative, intentional, and effective uses of technology to enhance student learning, uses that we hope point the way to the future of educational technology in college and university settings. In this episode, we're exploring that future by looking to the past. Leading Lines producer, Cliff Anderson, shares a fascinating interview with tech entrepreneur, Brian Dear, about his book titled *The Friendly Orange Glow: The Untold Story of the Rise of Cyberculture*. The book tells the story of PLATO, an experiment in the 1960's and 70's to see if a computer could teach people. In the interview, Brian Dear talks about the development of PLATO and its impact on the history of computing.

[0:53] He mentions a few names you likely know, like Douglas Engelbart, Seymour Papert, and Isaac Asimov, as well as a few you likely don't. And he discusses the origin and importance of things that we often take for granted today, like a display that responds as you type and the role of social connections in learning. This episode is a little longer than our usual, but if you have any interest in the history of computing, I think you'll find it really interesting. (music)

Cliff Anderson: [1:24] Alright, hi everyone. My name is Cliff Anderson. I'm associate university librarian for research and digital strategy at Vanderbilt University and one of the Leading Lines podcasters. And I'm really delighted to have with us today, Brian Dear. Brian is the author of the book, *The Friendly Orange Glow: The Untold Story of the Rise of Cyberculture* published in 2017 by Pantheon. And welcome Brian, thanks for joining us.

Brian Dear: [1:49] Thanks for having me.

Cliff Anderson: [1:50] Alright, so I want to just begin by sort of letting listeners know about the PLATO system, because to be honest, when I saw this book and it's a pretty massive work, more than 500 pages. And I came across it in a bookstore one day. And I thought I knew something about computing history. And I saw this and I thought, I don't know anything about this, that I need to read this book. And I'm embarrassed to say that that's a lacuna in my knowledge, but I'm afraid that you also wrote this book because many people don't have sufficient knowledge of what PLATO was and how much it accomplished. So why don't we just start by saying, what was PLATO? What did PLATO stand for and what was its goals?

Brian: [2:30] Well, it's an acronym and it stands for, stood for Programmed Logic for Automatic Teaching Operations. And it was begun in 1960. Back in the heyday of NASA and all the military acronyms and government acronyms, they were a big deal. I guess they still are. So the first thing you ever had to do when you started a technology project, was give it an acronym and start with the acronym and then I guess work back from there and try to figure out what the words actually would be. And they named it after the Greek guy and it was developed at the University of Illinois.

[3:14] The purpose was to essentially, it was an experiment, was to see whether you could build a computer system that would teach people and then test it to see whether it works. In other words, test the students to see whether they actually learned anything. And this was, say, a pretty bold idea. And considering the state of computing in 1960, it was a rather insane idea, but there was just such a fervor and excitement about trying new things and particularly really hard problems. And the PLATO project, if you just look in hindsight at it, was such a collection of unbelievably hard problems that many people in academia at the time had no interest in pursuing such a set of challenges because it was felt to be completely impossible. And yet they persisted and the PLATO project went on for many decades.

[4:20] And it's important also for people who, you're right, the vast majority of the public, you know, 99.9% of the world has never heard of PLATO, the computer system. And that was a shame because, you know, many people have heard of say, Apple and Xerox PARC and Microsoft and other famous projects that came along later. PLATO was just as important, just as big a deal and it got funded by the same agencies and supported and was well-known at the time. But it just somehow fell through the cracks of history and was about to be lost. And there are almost no magazines or articles or books or documentaries or pretty much anything about the PLATO system and the thousands of people who used it, developed on it and had their lives changed by it and met their spouses through it and, you know, all kinds of things. And it's such an essential component to the, not only the history of educational technology, but the history of social computing and computing in general, but also social, in other words meaning, how do people use computers to interact with other people? And so we can get into that more, I'm sure, with other questions.

Cliff: [5:50] Yeah, that's a really helpful way to start. And I think a lot of things that you said that we want get back to during our conversation. But let's just start with sort of the roots of PLATO. And as you laid out in the book, the roots really go back to B.F. Skinner and his desire to create a teaching machine. Can you tell us a little bit about how Skinner's behaviorist psychology, which I think a lot of us know about the idea of a Skinner box, influenced the PLATO vision of computer-based education?

Brian: [6:23] Well, it's a good question, though I would, in a way kind of challenge it because I actually think that it wasn't the behaviorism and the behaviorist psychology that influenced PLATO or the vision of what PLATO might be. It was the fact that Skinner had built a very primitive, or series, of primitive mechanical machines that he used to demonstrate with real students, college students, and kids as well in K through 12, that you could sit down with a machine and the machine would present you, I guess you could call it a morsel of information, you know, say in a tutorial basis or something. Or it might ask you to spell a word or something like that. And you would respond and it would give you immediate feedback and tell you whether you got it right or wrong.

[7:36] That was basically the essence of it. The problem was that it was all mechanical and it was slow and clunky and it just did not scale. And, you know, whenever you tackle anything in education, you have to think about how does this scale, because there's millions and millions of students. And, you know, you're not going to make much of an impact if it's just something that can only work for, you know, one student at a time or a small number of students, in one building, in one city, in one state. And that's about it. So Skinner's idea of using a machine to teach is a thing that I think really fired up at the imagination of the PLATO people, not necessarily his behaviorism.

[8:29] And you can also see that because the people that started the work on the PLATO system primarily came from engineering, computer science, physics, and other sciences and were not educators, except in the sense of being professors at a university. But they were not professors of education, is what I mean. And they typically weren't instructors in a school. Again, in a grade school kind of setting. That would come in time as more people join the project. But I think what really got them excited was the idea that, you know, a machine teaching people is an interesting idea. And by 1960, it was really obvious even outside of the

PLATO project, that if you're going to use a machine, you better use a computer because it solves a lot of the problems of scale and programming it and that kind of thing. And it's just the right way to go.

[9:37] It was really, really obvious. IBM had already figured that out and was doing little experiments a year earlier and or even two years earlier. So that's the essence. And then, you know the leader of the PLATO project, a guy named Donald Bitzer, who was only 26 in 1960, had just gotten his PhD, had just become a professor. He was an electrical engineer through and through, and not a psychologist, not an instructional theorist or anything like that. The philosophy he tried to maintain was one of, look, we're not sure exactly what theory of instruction or psychological theories or whatever are going to apply to using a computer to teach and instruct a student so why don't we just build a machine that is extensible and tries to stay agnostic to whatever philosophy you know, the instructor wants to use and let them be in charge of that rather than the machine inheriting and sort of dictating one school of thought. So while PLATO is often associated with being a behaviorist system, I think the years, as the years went by, it was pretty much proven that it was very agnostic. It tried very hard to not force an instructor who wanted to use the system for their students into any way of designing the instruction. So I hope that helps.

Cliff: [11:31] Yeah, that's extremely helpful. And I think it certainly comes across in the book, as we'll see later, just how freewheeling the environment was at the University of Illinois. And how many, you know, really interesting projects took shape because of that open-ended design philosophy that was apart of PLATO. So, but I wonder if we could actually, before we get to that, I want to come back to your comment about scale. And there's a passage, I think, on page 309 of your book, I wonder if you would read to us because it really talks, I think, about this movement from the individual teacher to a scalable computing system that could teach potentially millions of people. Would you mind reading that passage to us?

Brian: [12:16] Sure. Now this passage is from a chapter called "The Zoo," which is basically about a kind of a legendary classroom that was full of PLATO computer terminals rather than just rows of seats and desks and a blackboard and the conventional kind of design, it was a bunch of tables with these big box computer terminals on it. And it became rather legendary because of the gaming use in the evening. So when I mentioned the zoo in this passage, that's what that's referring to. So here it goes.

[12:52] *B.F. Skinner had cherished his experience with Mary Graves, his legendary grade school* *teacher, whom he had described as someone who listened to me, answered my questions and almost always had something interesting to say or a suggestion of something interesting to do. In the 1970's, for many students who discovered PLATO and became denizens of the zoo, the PLATO system, it's online community and its ever-growing catalog of lessons, notes files, games and countless other activities had become a different kind of Mary Graves. Instead of PLATO being the ultimate online teacher of the ultimate online academy, rivaling that of the Greek figure for whom the system was named, a digital place to come and learn about everything from anthropology to zoology. PLATO had become, for so many young people, a place to come and learn about PLATO, a place to learn about each other. The system itself was the thing. If you had a voracious appetite for learning, if you had somehow, somewhere in your life, learned how to learn and derived joy from the experience of learning then when it came to PLATO, the most fascinating learning of all was the system itself. The system and its online community had become a cyber proxy for that long lost someone who listened to you, answered your questions and almost always had something interesting to say or a suggestion of something interesting to do. And it's been that way ever since. In 2017-* which is when this book came out- *PLATO may be gone, but the phenomenon is still here only now, gargantuan in scale, changing humanity, changing the earth, no turning back. Who needs Mary Graves when you've got the net?*

Cliff: [14:42] I think that's a beautiful passage and I think it encapsulates so many themes in your book.

Brian: [14:49] It's funny you say that because when I interviewed B.F. Skinner, I was lucky enough to interview him in 1987, he told me the story about Mary Graves. He also wrote about her extensively in his autobiographies. But it dawned on me right away. It struck me right away that PLATO had become the Mary Graves of the digital age and today, the internet. And even more so, your smartphone has become the Mary Graves of that he had in person. And it's a sad tale in a way if you think about it, but it is nevertheless, our smartphones have become our digital companions in a way. And that's really what that whole passage was about.

Cliff: [15:45] And I think in fact in this section, you also refer back to the Isaac Asimov story, the fun that they had about thinking from the future back to learning in person with a teacher, if I understand correctly.

Brian: [15:57] Right. Absolutely. Yeah. I actually had written to Isaac Asimov and I had heard in the 1980's, while he was still alive, that he always, always responded to anybody who wrote to him. So I typed up because back then it was still typing, though I think I had a computer printer. I printed out a letter and I mailed it to him. And I could not believe my eyes one day when I got my self-addressed stamped envelope back, with my letter in it and a reply from him typed at the bottom of the page of my own letter. So he basically sent my letter back to me with his reply typed on it. And that's how he suggested that I look up that story. And so that's kind of interesting.

Cliff: [16:54] I mean, again, I think at the end, I want to come back to your process of writing this book because as you mentioned, this book took a while to write and you gathered from almost a lifetime of experience you've included in here, but returning to the two people that at least in my reading of your work, had most to do, I mean, there's so many people, of course, that had so many different roles in PLATO, but two that really stand out at are Daniel Alpert and Donald Bitzer. I wonder if you could talk about the role that those two had in sort of getting the computer-based education research lab, I think it's called Searle. Is that right? At the University of Illinois, Urbana-Champagne off the ground. What was it about their partnership that really sort of fomented the development of that lab?

Brian: [17:43] Well, Alpert ran a well-known laboratory called the Control Systems Lab that became the Coordinated Science Lab at the University of Illinois, CSL for short. And CSL, all during the fifties, was a classified military lab that did projects for the defense department. It had arisen out of the Korean War and kind of as a continuation of projects on a really massive scale, things like Manhattan Project and other kinds of major undertakings. And so they were doing things like radar and missile guidance and all kinds of crazy stuff that involves computers. So they got, they built their own computer at CSL called the Illiac in the fifties that became one of the first big computers in the world.

[18:42] But towards the end of the fifties, it was clear that they're going to, they needed to declassify the lab and start taking on civilian projects. And because of all the hoopla in the popular press about Skinner and his teaching machines and wasn't the future exciting and all this stuff, it became known to people at CSL, including Alpert, that there was this idea out there of using machines to teach students. And, you know, he thought that that might be something that they should explore at CSL as one of these new unclassified civilian kinds of projects. And long story short, he picked Don Bitzer, who was 26 years old. He had just gotten his PhD. He was, he had just gotten, I guess the, the lowest grade of professorship at the university, was an electrical engineering genius, well-known to CSL. He had been working there since, I guess his undergraduate years.

[19:46] And so he was totally gung-ho kind of personality. You know, loved difficult problems. This was perceived as being probably the most impossible, difficult problem of all because computers were so primitive at the time. But the amazing thing is that Bitzer and Alpert got along really well. They respected each other. And in a way, Bitzer was kind of like the startup founder in Silicon Valley. And Alpert played the role of sort of the mentor slash venture capitalist slash Board of Directors type chairman of the board or something. Someone who knew how to raise money, knew all the connections, had a huge network, had a great reputation out in the world. And Bitzer was this young whippersnapper, technical genius and wizard who attracted all kinds of other technical geniuses and wizards and engineering types around him.

[20:52] And so it was a great combination of personalities and that really helped sustain the whole PLATO project over many years. Alpert's role was one of protecting and blocking and tackling for Bitzer, making sure that he could stay focused on the technical issues. And Alpert would worry about where the money was coming from and if some bureaucratic aspect of the university or something where the government was frowning on something with the PLATO project or something. He would go fight that fight and let Bitzer stay focused on technology. And so that's how things went and that's why it's significant with those two folks.

Cliff: [21:39] Thank you and you know, what's interesting is, right out of the gate, this partnership developed really two, I mean, two fundamental innovations in computer science that continue to resonate to this day. And that is the creation of the plasma screen and also the development of timesharing. And one of the things that I think you note that again, kind of corrects our understanding of computer history, is that PLATO gets the credit for the development of the plasma screen. But it doesn't get the credit for the development of timesharing, which as you mentioned, typically is associated with work at MIT. So I wonder if you could talk about how those two innovations really sort of led to the initial implementation of PLATO. I know that I'm collapsing with a bit of history here too, because that's not necessarily the initial implementation.

Brian: [22:32] Right. Well, the funny thing is, one of the things you learn whenever you do computer history is that someone else always invented it first. And in fact, somebody in some garage or some basement somewhere probably did it before. Whatever it might have been, you know, whatever you're talking about it. For example, when I was researching whether or not PLATO, PLATO's chat and instant messaging features were ever done by others. Sure they were. It's just that, you know, they never had the traction and made any kind of impact until PLATO came along.

[23:13] And with the plasma display, what's interesting there is, what really blew my mind was that Douglas Engelbart, who is legendary in Silicon Valley for all kinds of innovations and is kind of considered one of the great pioneers and the fathers of Silicon Valley and everything. He got his start working on plasma, attempts at plasma displays and plasma arrays for memory in the late fifties. And this was a grid of tiny microscopic wires in an XY, kind of literally a grid. And then you could send signals on any x or y coordinate. And then wherever they crossed, that would, in theory, light up a bit of neon gas plasma, and it would stay lit, and that would be memory.

[24:10] And remember in the late fifties, RAM was still kind of exotic, weird thing for computers, random access memory. It was insanely expensive. And for the PLATO project, Bitzer realized in 1960 right away that he's going to need graphics display terminals for PLATO, since it's dealing with education and they're going to need to show pictures and drawings and graphs and charts and multiple text fonts and everything you can imagine. And if you used video RAM to do that in 1960, it costs $2 per bit. And if you think about, you know, I'm currently talking to you through a Macintosh laptop. That memory and that laptop at $2 a bit now would cost several, I think it's several, $100 billion or something like that. Because my machine has got 16 gig of RAM in it or something. So it is just impossible to scale.

[25:14] Memory was not going to work that way. And so the plasma project was a way to figure out, is it possible to use this kind of glowing gas neon dots in an array of XY wires? Not only as the display because it's basically pixels, but also as the memory. So when a student is looking at the screen, they're looking at the memory. They're looking basically at the RAM. And that was the breakthrough. Engelbart had tried to do it and he couldn't figure out how to do it. And a lot of projects around the country, both commercial and military and other academic projects were trying to figure it out and everybody was scratching their head and a lot of people were giving up saying it's just not possible to do. MIT, notoriously poo-pooed the idea and kept teasing Don Bitzer about it for years saying like you're never going to get that thing to work.

[26:13] And it's ironic because they did and it became very successful commercially. Plasma displays were used in, in all kinds of things like cash registers and calculators and later on televisions. Remember Fujitsu plasma vision TVs in the nineties. And then suddenly you'd go to airports and other public spaces and there'd be these huge flat wall screen televisions or monitors showing information or whatever. And that was all gas plasma originally now it's mostly LAD because that's cheaper. And plasma used a ton of energy so that, you know, there's that.

[26:56] With timesharing, it's another case of I think as computers started to catch on in the fifties and the very early sixties, any smart person and there were millions, thousands of them all around working on computers, came to the realization very fast that they needed essentially multiprocessing. They needed the computer to do lots of things simultaneously. And that kind of led to, you know, we need the computer to be able to not only do multiprocessing, but we need it to do multiprocessing for multiple users simultaneously. And that kind of led to the idea of timesharing. And the race was who could implement it and get it live and functional. And history has given the credit to MIT. But actually, the PLATO project had two simultaneous user version of PLATO running several months before MIT got their breakthrough. It's just that the university was very slow. University of Illinois was very slow filing the patent for Bitzer and company. And they took, they just, they weren't used to this kind of crazy file a patent and get it out kind of thing on a timeline that people expect when it comes to the Computer Age. And so MIT basically got all the fame and glory. But calendar wise, PLATO was doing timesharing before MIT's system. So that's how that kind of came about.

Cliff: [28:40] And you also tell a wonderful anecdote about a later version of the timesharing that was developed by someone, Andy Hanson, who was actually in his teens when he was developing it and then was a student at Harvard, went to hear Bitzer give a lecture at MIT, was sitting in the audience. They couldn't believe that the system worked in the way that he was describing it. And then Hanson said, and actually I'm the one that programmed it, which they couldn't believe either because the guy was 20 years old. I think it's just fantastic.

[29:13] So there's another aspect of PLATO that made it really successful. And this was the concept of the fast roundtrip. And I wonder if you could just tell us what that means, fast round trip. And then why it gave PLATO a decisive competitive advantage over other educational technology systems that were coming online at the time?

Brian: [29:32] Well, the fast round-trip. There's a technical sort of explanation and it's a term that I made up. It was not something that I ever heard, to my knowledge, the actual builders of PLATO use. But I needed to try to figure out some way of helping the reader understand a pretty set, a pretty complex set of concepts. And so basically, to me, it's really a design, in other words, a humanistic design feature of PLATO that is remarkable and sets PLATO aside from a lot of the other computer work that was going on. One of the things that Skinner had emphasized from his work in the fifties with primitive teaching machines was the idea of individual students could have, so the theory went, you know, a 100% of the attention of the quote unquote teacher. So in theory, responses would be instantaneous because the quote, unquote teacher has nothing better to do but attend to that student. And so the knowledge of the results would be immediate. And this whole sort of general idea of immediacy of a real fast response was floating around with the Skinner stuff. And I think that resonated well with Bitzer and the PLATO folks.

[30:57] When it came to using a computer to teach, if the student was presented a question on the screen like, let's say a simple thing like multiple choice, an ABCD set of questions and answers. And the student types, the letter a. You know, that signal would have to go from the keyboard through the terminal, through the, either the phone line or some other connection to the remote central computer, which would have to recognize that the letter a had been typed. And then send out instructions back to the terminal, to the screen to display what would render to the student as a letter a in the right spot on the screen. We take this completely for granted.

[31:43] But, you know, this was like revolutionary in 1960. This was real interactivity where you were interacting with a computer and it was right there with you. And there was no delay. And a lot of other computer systems could not do this. Or they had to create crutches like what I call, or what was called local echo, meaning that somehow the local terminal could figure out that you would type the letter a and send an a to the screen. You typically, there's a blinking cursor. And you know, a lot of old computers would have blinking cursors, but the problem was if when you type the letter a, the central computer didn't know that you typed anything yet because you hadn't pressed Enter or Return. PLATO didn't work that way. PLATO was live, every key on the keyboard was live. So the system was extremely sensitive. And this would turn out to be incredibly advantageous for doing not just simple kinds of question-and-answer stuff, but also real-time editing of code or text, if you're writing an email or something. It was much more like a microcomputer, even though you were on a central system with a terminal.

[33:10] And then when the gaming and the social aspects of PLATO came along in the seventies, that fast round-trip architectural design was so deeply inherent in the DNA of PLATO that it turned out to be a fantastic capability for doing advanced simulations or multiplayer games that were unbelievably addictive in super real-time. And you'd be slamming away at the keyboard. Even if the screen couldn't even keep up with you, the keyboard could. That was the crazy thing. The screen would pretty much be black because it couldn't keep up with the updates. But the keyboard and the connection to the central computer was fast enough that you could just be bam, bam, bam like a machine gun with your fingers. And it would keep up with you and do what you were telling it to do. So it was like trusting in the force in a way, you didn't have any information visually. But so yeah, it was a tremendous advantage.

[34:16] And even today we don't really have, we have the illusion of responsiveness because the micro computers are so fast. But when you're connecting to a website, there's still this clunky back and forth kind of stuff that's going on that is very unlike PLATO. And you see this even in educational uses of the web, where you try to take a tutorial or some kind of lesson through some educational website or something like that, or a quiz or something like that. And oftentimes, it's very old-fashioned where you're given a box to type in your input, but it's only after you press Enter that the whole block of text is sent to the computer and evaluated rather than character by character as you're typing. And PLATO was, everything was character by character. And so kind of a long answer to that.

Cliff: [35:26] No, that's fantastic. And again, another anecdote that I love from your book was that some of the students that use PLATO really distinguish themselves by improving their typing speed. And so there were people that could type, you know, 160 words per minute. They were sort of legendary because they just get those characters out faster than anyone else, which I love.

Brian: [35:46] And it was a thing in the PLATO world, this thing called Hall of Fame, where if you had the highest score, that was really important. And, for a lot of people, that was a way to distinguish yourself, just like making a big fool of yourself on Twitter today is a way of distinguishing yourself maybe. You'll get lots of followers and there are, you know, there's all these little cues that we have built-in to social computing now that we don't even really think about, but, you know, the number of followers, the number of likes, all that kind of stuff, all that kind of sentiment was, was apparent on PLATO in things like the halls of fame and the typing games were one of the most competitive areas. Where you'd go into a PLATO classroom and it will be full of people frantically typing to try to beat some legend who's in another state 200 miles away or something.

Cliff: [36:59] We've been talking a lot about the hardware side, but one of the things that also made PLATO really stand out was the software that was developed to enable regular folks and not necessarily professional programmers to write interactive lessons in other types of programs on PLATO. So this was called the TUTOR Programming Language, and I don't know if I got the fellow's name right, the lead designer, Paul Tenczar?

Brian: [37:25] Tenczar, correct.

Cliff: [37:26] Tenczar. But can you explain why Tenczar's development of TUTOR was so crucial to PLATO's success?

Brian: [37:35] Well, prior to that, there were lots of competing, what was called authoring solutions or languages or form filling out procedures that you would use to construct a series of pages that would present information to the student. And then the student would be given an opportunity to interactively respond to a question or multiple choice or text answer or something like that. Or be presented with some pictures on the screen and then identify something, or one piece of like, you know, a skeleton, identify the so-and-so bone or something like that. And they were all kind of difficult to use. And Paul Tenczar was increasingly frustrated that people were spending so much time filling out forms and doing all kinds of low-level grunt work just to create the equivalent of the age-old statement, "hello, world," which is often used in, you know, on a computer system. In other words, how hard is it to just display on the screen the text, "hello world."

[39:01] And you would be amazed at how hard it was in the 1960's to do that. You would have to sometimes on some systems know pretty much like assembly language. And the octal codes of the various individual characters, the h, the e, the l and all that kind of stuff. And so Paul Tenczar was just so fed up with that. He just wanted to be able to say, you know, at and then at a position on screen represented by a number or an x and y combination of numbers separated by a comma. And then the next line would be a command called write. And then you press indent or a tab, and then you just write the texts. So you just say, you know, like at ten ten, write "hello, world." And it worked. He designed the whole language to be like that. Just like, you know, you could make paragraphs of text appear here or there or whatever. You could display. You could put boxes up on the screen. You could draw circles. You could create all kinds of displays and everything.

[40:10] And the productivity gains were so immediate and so obvious that the TUTOR language just blew away everything else that was trying to compete as a solution for instructors who are creating their own educational lessons for their students. And TUTOR became the thing that, that you use to create any kind of material on PLATO. And so that happened so quickly, within a year. With the ongoing growth of the PLATO system and the staff and the funding and everything, the system staff was devoted increasingly to just developing and building out the TUTOR programming language.

[40:57] Now, from a computer science perspective, it's absolute heresy and they consider the TUTOR language an absolute joke. They're very snooty about it and that kind of thing. But it's actually very amusing to get a computer scientist and a PLATO person in the room and, you know, duke it out in terms of because they were designed for different purposes. And so, you know, the TUTOR language had a very specific mission which was to help people develop educational interactive lessons and simulations and things like that. And for students on the PLATO system period, that's what its mission was. It wasn't to do data processing for a bank using the COBOL language or Fortran, or it wasn't to crunch numbers for, you know, accounting departments at huge corporations or anything like that. And there were other tools for those things. And over time, things optimized to be the best tool for the job. TUTOR was designed to be the best tool for developing material on PLATO. And so that's how that came about.

Cliff: [42:17] Yeah, that's really helpful. And I think today about communities of practice that built up around like the R programming language, for example, which was developed by statisticians, as you know. But also, you know, from a computer science perspective, sometimes people look at it askance because it was developed by folks other than computer scientists. And that means it has different design decisions baked into its core. And I think that sort of dynamic has worked out in other areas as well. So it's interesting to hear about how that worked for TUTOR.

[42:51] I want to come back to the community that build up around TUTOR because I think one of the things that's really wonderful about reading your book is you share so many anecdotes about how TUTOR affected people's lives and really shaped the direction of their lives at a young age. And in particularly of course, people growing up in Urbana-Champaign, but also in other places where there were PLATO systems setup. But particularly for this group that was maybe going to high school in Urbana and would wander over to the building and discover PLATO. How was it that virtual community sort of immersion? I know you can't do all the stories, but can you just say a word about the way in which the Searle was open to kids coming in and sort of fostered their development in this kind of ad hoc ways that wasn't really part of the formal program, but just had such a tremendous effect on so many people's lives?

Brian: [43:45] Well, again, this gets back to Alpert and Bitzer. One of the things they absolutely believed in was if you find smart people, invite them in and help them join the project and help get the mission to be successful. And so it was an extraordinary laboratory in the sense that almost any other lab in the United States or probably anywhere in the world, would have had guards and keys and locks and, you know, administrators and a bureaucracy that would keep the public out. But with PLATO, and this was a philosophy that Bitzer and Alpert had from day one. If you find some bright 12-year-old is wandering through the halls, get them to sit down in front of a PLATO terminal and ask them a few questions and see if, see how bright they are and how curious they are. And if they really light up and get excited and ask great questions and then give them a problem to solve and tell them, you know, you work at that and I'll check on you tomorrow or something like that.

[44:59] And this was done constantly. And word would get out that the PLATO lab was really cool. And people were developing all kinds of cool things and you wouldn't get chased away, unless you were really obnoxious. But it was essentially unheard of that there was a laboratory like this that had become its own building and had a staff of like a 100 or so. And they were very open and enthusiastic to anyone wandering in off the street. And so kids in high schools all over Champaign-Urbana heard about this and would come over and check it out. And if you had, if you were kind of a techie at heart or a geek or whatever, as it was called later, you might find that this is your crowd and they're doing really cool things and in fact, word would get out that, you know, if you really knuckle down and devoted some time and solved problems and everything, you could get a job for even higher than minimum wage, working on the system and have fun doing it and learn tons of things that would help with career in the future.

[46:16] And so this, the other thing was, of course, since PLATO was developed for students as the ultimate user of the system, they were constantly in need of students to test things out. And so instructors and developers would, if they saw some kid wondering around, go "hey, sit down at this and take this lesson and tell us what you think and, you know, and does it make sense to you?" And so, you know, there was a benefit, there was both a technical benefit, as well as an economic benefit because kids were cheap. They were delighted to do work for minimum wage. And so you didn't have to hire some post-doc, a postgraduate student who might charge thousands of dollars to do a project when a kid of fifteen or something might do it for a dollar an hour or something like that.

[47:09] Because of this and because word got out that you could use PLATO and the TUTOR language to build games, that was it. That and the notion of using PLATO to chat and yack and flirt and everything else with other users on the network. And to build applications that could support that kind of thing, like a chat room or multiple chatrooms. That all started popping up in 1973. And in one year, one 12-month period, basically, teenagers transformed the PLATO system by building a fully-fledged online forum system, a message board system. Email, instant messaging, screen-sharing, chat rooms with even private rooms, password protection if you wanted, and tons of multiplayer games and all kinds of other things, all in 12 months, it was one of the most remarkable explosions of creativity.

[48:23] And that you wouldn't see until the nineties when the web exploded, largely because for a similar kind of reason in the sense that a browser allowed you to view source. And you could go in and see how an HTML page was built. And that inspired millions of people to build their own HTML pages. And I've always argued that that's really how the web exploded and took off. And so the similar kind of thing was happening with PLATO and the kids in the early seventies.

Cliff: [48:57] And I think that's another really interesting aspect of your book is that in a way it's a history of PLATO, but it's also a component history of video games. Because there's so much description of the games that the kids made during that period. And I wonder if maybe you could talk a little bit about the way that I think, as you describe it, Don Bitzer began to realize that if PLATO was going to succeed, it had to actually like bottle the energy that was coming out of games. Thye couldn't simply rely on the, maybe the stuffier side of it, which was educational technology. Or it had to at least think about how those two were intertwined. There was a meeting you describe of a lot of educational theorists, technologists, I believe Seymour Papert and others were there in which they came to a head about this conversation. And it seems to me that Bitzer really was onto something, but it didn't quite work out as we might have hoped. But I wonder if you can comment on the role of gaming in particular moving the system forward.

Brian: [49:58] Well, it's not just gaming. You could call it social gaming because social is to me the things. It's the fact that you could, you were aware that there were other people connected to this network as well, and that you could interact with them both in conventional ways, like with chat and stuff or you could interact with them through the imagination of a game where you're flying a spaceship or an airplane or something like that. But there's always messaging between the gamers. And I really think that at the end of the day, it was the messaging and the games, just like it is today. I mean, video gaming is a vast industry on a scale, I believe, order of magnitude bigger than the movie industry, which now of course with COVID is kind of kaput.

[50:53] But you know, the thing about the gaming even now is all the yacking and the talking and everything that goes on between the players. And that was absolutely the case with PLATO. So it's not just that they were video games, it was that they were social games pretty much from the start. And in addition to the gaming, there were, again, the notes files, which were the message forums and that kind of thing exploded in terms of the number of different topic areas that there were notes files for. People would yack about movies and music and sports and television and politics. And, you know, then those things would get so big, they'd have to branch out to be like well conservative politics and liberal politics and this and that. Football would have its own notes file and basketball and that kind of thing because the volume had just gotten too big. And so all of this was happening simultaneously.

[52:09] And so I think that's the really significant thing with PLATO is that, you know, and Bitzer couldn't help but see this. He saw that the future was going to be people living in front of screens. And they were not just going to sit there and take lessons on Zoology. And they weren't going to do what the corporations imagined, the homemaker sitting in the kitchen with their computer terminal looking at recipes. You know, that these were just, it's often hilarious to look at the ads from computer magazines of the seventies and eighties to see what corporations imagined people would be using the computers for. They simply did not understand what Bitzer saw, which was that people are going to sit down and use computers to interact with other people, period. That's what it's all about. It's all about social. And even the microcomputer companies, Apple and especially Microsoft, were clueless about social uses of computers. I would argue they didn't know squat about this stuff. And were very reluctant to sort of go in that direction until, you know, maybe the late eighties or nineties and then, you know, I think went into it kicking and screaming. Now of course they claim they invented at all, right?

[53:43] But it wasn't that way. The social dimension of using computers was not something that you used a microcomputer for. It was something you needed a network for. You needed a central system or a timesharing or something like that. You needed other people to be able to communicate with others. I mean, without the network, you don't have the social. And micros were desert islands without any social. And that's why what you saw as the original historic applications of microcomputers are spreadsheets and word processing, and number crunching and that kind of thing, very lonely, and non-social kinds of stuff.

Cliff: [54:30] Yeah. I mean, I'd love to continue the conversation along those lines because it does reflect my own experience growing up with an Apple II. Looking back at it, I mean it was a fascinating computer to use. And you talk a lot about the Apple II. And I did have a modem. It would communicate very, very slowly with bulletin board systems, but it was definitely not a social computer in any way like, like PLATO was.

Brian: [55:00] Right. But it's crucial that, you know, what a transformation the modem had and bulletin board systems in the eighties had to micro computer users. They realized like, wow, we're all out there and we can all actually connect with each other through these BBS systems. And then of course, there were early things like CompuServe and Prodigy and the Source. And those were the early kind of computer networks that people belong to. And they had an important role to play, to educate people that, you know, the real killer app for computers is people. And I would argue it always has been. And PLATO proved that.

Cliff: [55:49] Well, I want to talk a little bit about some of the places that PLATO missed opportunities along the way. Because we've been talking about a lot of the things that it got right or introduced into the computing world and referred to some of these like email or the original types of blogging systems and things like that. But I'm wondering, there were two places I think that in my reading of your book that PLATO missed opportunities and probably many more like any system. But one is it didn't connect with the growing network that was taking place at the time, which was much larger then, which was the ARPANET, right? And the other is, as we were just talking about, it didn't necessarily find a good place for the microcomputer in its architecture. And there was a kind of famous disagreement that you highlight with Alan Kay, in terms of where emphasis ought to be placed on the microcomputer or on the mainframe systems that were networked. And I wonder if we think about those two, would you call them failures or at least places where PLATO didn't make the connections? How did that inhibit its growth as we got sort of towards the late eighties and early nineties?

Brian: [57:09] Well, one thing is regarding ARPANET, it's really funny. I have to tell a quick anecdote. There is a thing in the world of the Internet called the RFC or the request for comment. And it was essentially a set of documents that were numbered starting with RFC1 and then RFC2 and then RFC500 and on and on and on. And this was the way that the ARPANET creators, which share proposals and ideas that would emerge and become standards for the communication protocols and the formats for things like email. There's a famous RFC822, which is the basis for all of our email in the world. So there's gazillions of these RFC documents. And there are repositories all over the internet of the RFC documents.

[58:09] And there was one called RFC600 that I started poking around, around the year 2000 because I'd heard about it and it had to do with PLATO. And it was like, oh, I must find that. And then I couldn't. It turned out that, and this is so perfect, that it just goes to show how obscure PLATO had become that nobody cared that there was one RFC document missing and all of the global repositories, even at famous universities and things like that, research universities and stuff that had all the RFC documents online. You could go get them and download them and everything, except RFC600, which was missing. And so I found out through looking in books that mentioned it, that it was about connecting an Illinois plasma terminal, they didn't even call it PLATO to the ARPANET. And it was a project done at UC Santa Barbara around 1973.

[59:15] And the reason they wanted to do it is because they heard that's where all the cool games were. And so they, the ARPANET people at UC Santa Barbara, tried really hard to connect a PLATO terminal with the gas plasma display and a touchscreen, we haven't talked about that much, but all the fancy stuff and get that over working so that you could sit at UC Santa Barbara in a room with a terminal and connect over the ARPANET to the PLATO system at the University of Illinois, which likewise had an ARPANET system, a box as it were, in the same building as PLATO. And so they tried really hard for months and months and months and they got it kind of working. It's just that it was so ridiculously slow that it was unusable.

[1:00:17] And the reason it was slow is because, and this is very telling, is that the ARPANET did not have the philosophy of the fast round-trip at all. And in fact, because it's all packet networks and so they're very slow and it can't handle instantaneous responsiveness. It couldn't handle me typing ABCDEFG kind of thing in real-time, really fast. And having those letters go across the network and fly back to display on a screen. ARPANET was not designed for that. It was designed for you would type ABCDEFG or whatever in your local terminal and then you'd press enter and it would send a packet with that string of text to it, which makes a lot more sense. It's much more efficient.

[1:01:08] And so from the ARPANET architectural perspective, they thought that was the way to scale so that you can have millions of computers communicating over a network. And they were right. And the internet evolved that way, and the web evolved on top of that in the same way. And over time, as the industry saw that this was all a good thing, they applied all kinds of R and D to figure out how to make the network really fast. But in its essence, it's still a packet network. And it's not the fast round-trip. It is effectively a fast round trip because the network is now so fast and we all have, you know, megabit downloading to our homes and stuff like that. So that's the big difference with ARPANET.

[1:02:01] So I wouldn't call it a failure. It was, again, PLATO was designed for PLATO. It was not designed to run on other networks. Now, you could say that that was a real shortcoming. And in the long run it was a thing that blocked PLATO from growing in some ways, I guess, but eventually, actually they solve that problem. And in the eighties, Control Data Corporation, got PLATO working over a packet network. And then by the nineties, the NovaNet system, which was the kind of one of the successors of the PLATO's system. It was pure PLATO basically. And it was still developed at the University of Illinois or by people at the U of I or in that area. They got NovaNet running on the internet. And they used applications that they wrote for Mac and Windows so that you could interact with NovaNet, which was essentially PLATO, through the internet. And so eventually that problem was solved completely.
And NovaNet grew and scaled onto a scale that was vastly bigger than PLATO because they were able to conquer that technical problem. So hopefully that helps.

Cliff: [1:03:22] So as a way of wrapping up, I just want to ask you about your process of writing this book because I think one thing that's really amazing about it is, if I understand correctly, it looks like you started research for the book in the eighties, while in some ways you had a connection to the PLATO system. And then completed the book in 2017, just as the last sort of version of the system went offline. And so this book spans roughly a 30-year period. And I wonder if you could just talk a little bit about your process of writing it. Because I have to say from my personal side, I'm just really impressed that you followed through and you got the project done. I'm so glad that you did it because it does document this really incredible history.

Brian: [1:04:04] Well, yeah, I got my start on a PLATO system as a freshman at the University of Delaware in 1979. I had seen the system while touring colleges the year before and was just blown away by what I saw. I've always had a lifetime interest in music and I was wandering through the music building and I saw this classroom where the lights were turned off, but there were all these people in it and they had headphones on and their faces were lit and glowing orange. And they were all looking at these gorgeous computer displays that were showing graphics of piano keyboards with gorgeous musical notation and musical notes, a treble clef and bass clef and all this stuff. And the notes were there. And they were reaching out and touching the screen and you could tell they were playing music and I could just not believe it.

[1:05:01] So I always remembered that about Delaware. And when I returned to Delaware, actually attended there, the first thing I sought out was that room and then I found the PLATO system and I got a job on PLATO almost immediately after learning the TUTOR language, they hired me. And so I actually worked on the system for a year or two and got completely hooked by it and realized computers were my career and that kind of thing.

[1:05:31] And then several things happened in the early eighties. First there was a book, *The Right Stuff* by Tom Wolfe that came out about the history of the astronaut, the Mercury astronauts, which was mind-blowing. And that was followed by a movie of that book. And then there was a book called *The Soul of the New Machine* by Tracy Kidder, which was one of the first really compelling computer books. It was this journalist spent a year, day by day, in the building with a team at Data General who were scrambling to try to get this computer system out. It sounds very boring as a topic, but he made it absolutely fascinating. And it was like the first great sort of story about what it's like to work in one of these fast-paced, crazy tech companies.

[1:06:22] And then the book *Hackers: Heroes of the Computer Revolution* came out by Steven Levy. And that was the whole history of, you know, MIT and timesharing and all that. And then a lot of stuff with Apple in Silicon Valley and nothing about PLATO. It was never mentioned. And that I just kind of, I thought, you know, there's a whole other volume to the *Hackers* story. And that book became legend and has kind of sealed the fate of what the mythology of computer history is and everything. And there was nothing about PLATO. And that got me fired up to do something about it because I felt by 1985 that PLATO system was starting to die. And you know, what's going to happen to all this rich history and all the innovations and everything?

[1:07:16] So I actually went out to Searle in 1985. I drove out, spent about a week there and interviewed anybody and everybody I could. And I still have those tapes. And, you know, what I discovered is I talked to one person and they would mention 25 names in the interview and

I’d have to track down 25 people. And then those individuals would all mention 25 names and this thing just exploded. And I realized that there are no books, no real articles about the human side of the development of PLATO, the people side. Who were these people? What were their personalities and their motivations and stories, and how did they meet and all that kind of stuff? There was nothing of that. And yet that was all being documented in the rest of the computer world, like it had been done in *Hackers* and all that. So I got very interested in that.

[1:08:14] But I realized I’m just going have to keep talking to people to capture and rescue the story from being lost. Because nobody, not even a historian, had any interest in this field, largely because they didn’t even know about it. I mean, this thing was literally on the precipice of being lost. And so that really kept me going for years. And by 1996 I actually launched a website. And what I decided to do was list out the names of everybody that I had managed to find. And then I posted a series of questions for anyone who had anything to do with PLATO, please get in touch, et cetera. And this was just around the time where search engines were starting to happen. And people, of course, learned that the first thing you do with a search engine is type in your own name and see what the net has to say about you. And people would type in their name and they’d find themselves listed on this PLATO page. And find out that there’s questions and some of the questions were does anybody know where so-and-so is in that kind of thing?

[1:09:18] And this was kind of how I did the research. Well, that website got a ton of traffic and led to me interviewing about 1000 people. And I manually transcribed all of those interviews. There were 7 million words of interviews and it took years to do. But then I had this gold mine of a digital archive of interview transcripts that I could do instant searches on and everything. And from that emerged, you know, the stories and whatever. And I also realized during the eighties, just to step back, that I needed to talk to people outside of PLATO, which led me to interview Seymour Papert and B.F. Skinner and pretty much anybody who was anybody in the field. Because I wasn’t really sure where the story was going and so I just kept letting it go.

[1:10:12] And I was doing this all on my own. I had no funding, I had no publisher, no agent. You know, there was no backing. I tried to get backing from the Sloan Foundation. And it turned out this was actually relevant is that Sloan Foundation, at the time, was very much an IBM shop. They had a lot of IBM money. The people that were running it were pretty much loyal to IBM. And they hated control data and anything to do with control data. And PLATO had a deep tie with control data. And control data was the enemy. And I think that's why the Sloan Foundation wouldn't fund my project. They had a whole series of computer history books, and mine was perfect for it. But I think there was just politics, you know.

[1:11:02] And I would find this constantly with the PLATO research is that there was tremendous amount of bias against like, "oh, you're working on the PLATO project, that's of no consequence," trying to downplay it, whatever. And, you know, it's a shame because mostly it was just out of ignorance. People were not familiar with the system and therefore, did not know how tremendously significant, from a historical perspective, it was. But it took a lot of years to come up with a narrative structure and organize the book into three parts.

[1:11:37] The reason there are three parts of the book is because I fully realize, I fully expected that the book was going to be three volumes. And I got a nice quick scolding from Knopf Doubleday, ultimately the publisher and they're with the Pantheon division, which was an imprint basically. But I had Knopf Doubleday editors and stuff. And they were the only publisher in the world that loved the idea of a long lost just on the precipice computer history that is about to be lost. If we don't publish this, it's gone. And so they were tremendously supportive. Though they told me right away, forget the notion of three volumes. It's going to be killer enough just to get one volume out. And so I shrunk the book down, chopped out many chapters, some of which I still have and plan to get out.

[1:12:35] In fact, I did release a chapter that never made it into the book, last year, or maybe it was earlier this year, about Roger Ebert, the famous film critic and his connection to PLATO. And if you Google that, you'll find it. Roger Ebert grew up and went to high school in Champaign-Urbana and was a budding journalist and wound up covering PLATO for a 1962, two-part series of articles in the news gazette of Champagne-Urbana. And you know, it took me many, many years to find those articles. And I actually tried to track down Roger and talk to him about it and stuff. But there's a whole long interesting twist to it, including a TED talk. And the very, very last thing Roger Ebert ever wrote before he died was a very short science fiction story. And it includes mention of the PLATO system. And I just think that that's pretty remarkable.

Cliff: [1:13:42] That's fantastic.

Brian: [1:13:43] Yeah. So that is out there. It's on the net. But anyway, so yeah. The book took a long time to do, but, you know, it finally all came together. I decided consciously that this was not going to be a "great man" kind of book. Many computer histories or even, you know, technology or business histories and stories or narratives or biographies focus on one guy who's like the brilliant visionary, et cetera. And with PLATO, the book is really a biography of the vision that infected so many people rather than just one person. And, you know, Bitzer had really directly no involvement with the creation of social and gaming and all that stuff in the seventies. But it was what he had fostered and supported and allowed and pushed with the notion of the fast roundtrip and stuff that enabled that kind of stuff to flourish. And the fact that he did not shut it all down because it was not strictly part of the mission is one of the great aspects of his vision. And that he was never, he was so open-minded that it enabled this kind of stuff to take off.

[1:15:15] And, you know, there are still people in the computer industry today, who were those kids that stumbled into the PLATO lab in the seventies and wound up in a wildly successful career in the computer industry. They're still out there and they work at Google, Apple, Microsoft, you name it, they're everywhere. And so, you know, and they would regale their colleagues with stories of PLATO, who of course would all roll their eyes going, "I have no idea what you're talking about." And that was another motivation for trying to get this book done is that hopefully this book would serve as a way to get PLATO into the conversation, finally, rather than being this oddity that is still outside of common knowledge.

[1:16:06] And even to this day, it's still a battle to try to get recognition for PLATO, the Computer History Museum, which is kind of like the Smithsonian of museums for computer technology in Silicon Valley, still does not have a PLATO exhibit. And you know, I've been pestering them since 2003 to do something about that. So there's still a lot of work to do to try to get more exposure. And my other hope is that there will be numerous other books written about PLATO. It's a vast goldmine of, of knowledge and stories and developments and intrigues and all kinds of stuff. And just like there have been dozens of books written about Apple, or Xerox PARC or Microsoft, or you name it Google, Facebook, Twitter. They all have books out. You know, I think a shelf full of PLATO related books is certainly a worthwhile endeavor. And I encourage others to keep at it. And so there are numerous perspectives and I could barely fit what I could fit in 640 pages. There's thousands more out there waiting to be printed.

Cliff: [1:17:24] I think your admonition that we need to know the history of educational technology is really well received and I know our listeners will appreciate it. So I just want to thank you so much for this conversation. I just learned a tremendous amount, both from our talk and from the book. And I will encourage all of our listeners to pick up their own copies. So thank you, Brian Dear. I appreciate it.

Brian: [1:17:46] Thanks so much for having me on and I'm just very honored and pleased that you enjoyed the book and encourage people to check it out and let me know what you think. (music)

Derek: [1:18:01] That was Brian Dear, author of *The Friendly Orange Glow: The Untold Story of the Rise of Cyberculture* from Penguin Random House. Thanks to Cliff Anderson for that really fascinating interview. In the show notes you'll find links to Brian Dear's website and Twitter account, as well as to that Roger Ebert's story he mentioned.

[1:18:19] Listening to Brian's talk about his first encounter with PLATO, seeing students in a dark room making music on computers using glowing touchscreens, I couldn't help but think of Ada Lovelace, the 19th century English mathematician and writer who is widely regarded as the world's first computer programmer. Her colleague Charles Babbage, designed but wasn't able to build the first programmable computer called the Analytical Engine and Lovelace wrote the first program for that computer. She also famously imagined the kinds of things that computers might do one day. In her notes about the engine, Lovelace wrote, "supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of mathematical expression and adaptations. The Analytical Engine might compose elaborate and scientific pieces of music of any degree of complexity or extent." It took over a century for that to happen, but Lovelace had the right vision.

[1:19:22] Leading Lines is produced by the Vanderbilt Center for Teaching and the Jean and Alexander Heard libraries. This episode was edited by Rhett McDaniel. Look for new episodes when we publish them. I'm your host, Derek Bruff. Thanks for listening. (music)