# Interaction at Lincoln Laboratory in the 1960's: Looking Forward – Looking Back

**Organizer/Moderator**: **William Buxton,** Microsoft Research, Cambridge. bill@billbuxton.com

**Panelists**:
**Ron Baecker,** Dept. of Computer Science, University of Toronto
**Wesley Clark,** Clark, Rockoff and Associates, Brooklyn, NY
**Fontaine Richardson,** Private Investor, Carlisle, MA
**Ivan Sutherland,** Sun Fellow & VP Sun Microsystems
**W.R. "Bert" Sutherland,** Past Director Sun Microsystems Laboratories

**Discussant**: **Austin Henderson,** Director, Research Strategy, Advanced Concepts & Technology Group, Pitney Bowes

## ABSTRACT
The activity centered around the TX-2 computer at Lincoln Laboratory in the 1960's laid the foundation for much of HCI. Through the use of archival film footage, and live presentations by some of the key protagonists, this panel is intended to contribute to a more general awareness of this work, its historical importance to HCI, and its relevance to research today.

### Keywords
Human-Computer Interaction, Interaction History; Lincoln Laboratory, TX-2, Computer Graphics History.

## INTRODUCTION
One of the most important and influential birthplaces of HCI was the work on interaction and graphics centered around the TX-2 computer at MIT's Lincoln Laboratory in the 1960's [25]. For example, it is hard to imagine the innovation that happened at Xerox PARC in the '70s having been possible without the foundation that Lincoln Labs provided. At a personal level, as a graduate student of Ron Baecker in the 1970's, I was the direct beneficiary of this legacy. It is no exaggeration to state that the work at Lincoln Labs was fundamental in shaping my own career.

Nevertheless, much, if not most, of the work there has slipped from our collective consciousness, with Ivan Sutherland's "Sketchpad" [22][23] system being the notable exception

The reputation that Sketchpad has garnered in the HCI and Computer Graphics community is well deserved; however, it was one among many projects – almost all of which are worthy of our attention, even (and perhaps especially) today.

**Figure 1: The TX-2 Computer at Lincoln Laboratory, designed by Wesley Clark**

But it is not just the projects or the individuals that made Lincoln Labs special. It may be that its most important innovation was the culture that it developed – a culture that not only permitted these projects to happen, but provided a strong catalyst to their doing so.

Lincoln Labs was formed in 1951. It was funded by the US Department of Defense, and administered by MIT. While much of the research undertaken at the lab had to do with air defense, there was also a place for exploring new and emerging technologies of interest.

The technological foundation for the work in the 60's was the Whirlwind computer, developed at MIT, and commissioned in 1951. Whirlwind set the direction of the lab in that, unlike most computing at the time, it was interactive, rather than batch, and it was the first digital computer that could display real-time text and graphics.


*CHI 2005,* April 2–7, 2005, Portland, Oregon, USA.
ACM 1-59593-002-7/05/0004.

Wes Clark, the architect of the TX-0 and TX-2 that followed Whirlwind as the research platforms in the lab, was passionate about computation being rich, intimate and interactive. His designs were driven by this sentiment, which he expresses quite clearly in the following:

> *… both of the Cambridge machines, Whirlwind and MTC, had been completely committed to the air defence effort and were no longer available for general use. The only surviving computing system paradigm seen by M.I.T. students and faculty was that of a very large International Business Machine in a tightly sealed Computation Center: the computer not as tool, but as demigod. Although we were not happy about giving up the TX-O, it was clear that making this small part of Lincoln's advanced technology available to a larger M.I.T. community would be an important corrective step* [6].

That one brief passage captures, as well as anything that I have read, the concern with, and dedication to, the environment in which students and researchers worked. Having built the TX-0 as an interim learning step, Clark, along with 8 other colleagues, did the engineering that resulted in the TX-2 in one year! [5][10][11][17]. (It was commissioned in June 1958.) Their being able to do so is another indication of the nature of the teamwork that pervaded the lab. It is also a measure of their ability to capitalize on what they had learned from earlier work undertaken there. Things like the TX-2 or Sketchpad do not come out of a vacuum.

As is wonderfully illustrated by [14] and [15], many of the foundations for graphical interaction had been established through experience with Whirlwind and the TX-0, and Clark was very clear about exploiting everything that he could learn from it. The hardware that resulted not only provided the platform for the research discussed in this panel, it was also the basis for the first computers from Digital Equipment Corporation.

> *Lincoln built the TX-0 computer to test the use of transistor circuitry and a large core memory and then the TX-2 for large-scale computing experiments. Both had a point-plotting display with I0" x I0" area and IK point resolution, light pen, camera, switch input, and abilities to interconnect arbitrary i/o devices. Hardware innovations of the TX-2 Include: addressable magnetic tape for a filing system; The Lincoln Writer, a typewriter for engineering/scientific use; and the TX-2 multiple sequence operation for rapid context switching. While the TX-2 was initially a personal computer, it operated under control of a timesharing operating system by the mid 60's. The TX computer circuitry was virtually identical to the logic and laboratory modules that DEC sold in its first four years, prior to the introduction of the PDP-1 in 1961* [3].xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

As Ivan Sutherland said in a 1989 interview, "…it would have been difficult to do the kind of things that I did with graphics at that time without the very strong facilities that were available at Lincoln Laboratory" [12]. Likewise, Ron Baecker has described to me how "the LEAP high-level ALGOL-like associative processing and graphics programming language of Feldman and Rovner [9], … made work like Genesys … incomparably easier." In short, the lab had been carefully crafted to provide the intellectual, cultural and technological heritage and culture within which great ideas could be cultivated and thrive, internally and externally.

**Figure 2: Ivan Sutherland using a lightpen to interact with CAD drawings using Sketchpad**

A prime purpose of this panel is to bring together some of the key pioneers from the 60's to show their work and tell their stories, both personally, and through the archive of 16mm films that we have that document their work in action.

Obviously, in the context of this panel, we can't bring together all of the contributors from that time. Missing, for example, are people like Larry Roberts.

Around 1966 he and Tom Marill, of System Development Corporation (SDC) in Santa Monica, linked together Lincoln's TX-2 computer with SDC's Q-32/PDP- 1. This demonstrated that (an albeit low bandwidth) packet-switching network could work across the country, linking two different computers running two different operating systems, thereby making a key step in the evolution of what became the Internet [16]. As well, in 1972 he wrote the first email management program, "RD". What is less well known is that he is a major contributor to the birth of 3D graphics as we know it. He introduced the use of homogenous coordinate systems to represent 3D transformations, which is still the mathematical foundation of the graphics in today's video games, visual effects, visualization and animation [19]. He evolved the first effective hidden line removal algorithm [18], and he did some of the first work in computer reconstruction of 3D geometry from photographs [18]. In addition, he was the inventor of perhaps the first interactive device for 3D input, the Lincoln Wand [20].

The point here is not to single out Larry, who is one of many, or even the members of the panel. Rather, it is to highlight the depth and richness of the team and the environment as a whole.

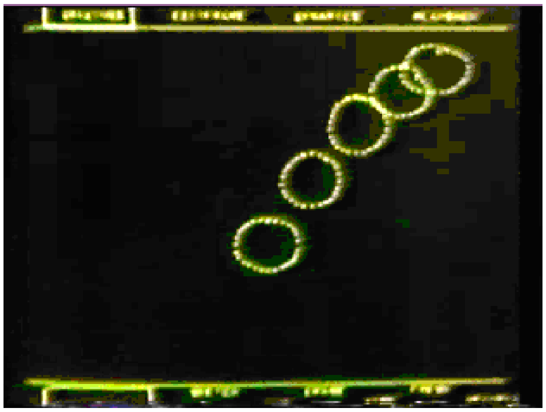

**Figure 3: Part of an animation of a bouncing ball taken from Ron Baecker's Genesys program. In this example, the balls are drawn freehand using a tablet, and each ball represents a different "frame" in the animation. This overlaying of frames is called "ghosting" today.**

Furthermore, my purpose goes beyond just helping create a better understanding of the origins of our discipline. My personal view is that much of this work is not just unknown, its potential is still largely underdeveloped. I firmly believe that some of this work is as relevant today, and perhaps more, than it was when it was first done.

Finally, and perhaps most important, I believe that there are aspects of the research environment at the lab that are worth taking note of in terms of how we design our own group cultures. Ivan Sutherland has said:

> *It seems to me the secret to success in research activities is to pick something easy enough to do. It's very nice if you can find something that you think is easy and everybody else thinks is hard. That's wonderful.*[12]

The other side of this coin is what is "easy enough to do" depends a lot on the environment where you work. What tools are available? Do you have the financial support? Who else is there that can "fill in the gaps"? Are they open to doing so? Sketchpad was only "easy enough to do" because of the prior work of Wes Clark and John Gilmore [13][15], among others, and furthermore, the fact that that work took place at that particular place where Sutherland was, Lincoln Labs. Likewise, that that work had taken place there was part of the reason that the Sutherland brothers were attracted to Lincoln Labs in the first place. Research environments function as magnets to talent as well as catalysts to ideas.

This lab was doing something right, and did so over a sustained period of time. The work in the 50's paved the way for that in the 60's. There is something to learn from this culture. From my reading of the literature, and conversations with some of the protagonists, much of the productivity of Lincoln labs stemmed from a combination of cultural factors, including:

- Freedom and encouragement to pursue ideas
- An ethic of supporting the full "food chain" of an idea: concept to refinement to building to using, and then iterating.
- Absence of constraints from a narrow predetermined agenda
- No constraints regarding the creation of short-term pay-offs. Sketchpad, for example, did not emerge, for example, until 12 years after the lab was founded.
- The building "of", as well as "on", a tradition of work over a sustained period.
- Attention to creating a physical, technical, social, cultural, and intellectual environment to foster innovation as well as attract the best talent.
- A strong concern and ethic around mentoring people and helping maximize the development of their potential.
- Making a priority of providing (by buying or building) the right tools for the job.

Perhaps the greatest potential take-away from this panel are the insights that the panelists can provide about these types of things. Their collective work certainly lends credibility to anything that they say around this topic.

In summary, my personal hope is that this panel will bring a renewed interest in the important but unsung contributions of the work, people, and culture of Lincoln Labs. As well, I hope that it will give us some cause to recalibrate how and where we set the bar for our own research. We have a lot to learn from what went on around the TX-2. I hope that this panel makes some meaningful contribution to our doing so.

### PARTICIPANTS:

*Ron Baecker* built one of the world's first interactive systems for real-time animation, *Genesys* [1]. The system had features that are still not available on the most "advanced" animation systems, despite being implemented in 1969 on a computer (the TX-2), which had less memory and processing power than a modern cell phone or *iPod*. More to the point, features of Genesys, such as "p-curves," are of great relevance in the support of things like "chalk-talk" dialogues. This is something that will be of increasing importance with the emergence of large format wall-mounted displays. Yet, this aspect of Baecker's work has essentially been ignored since he first demonstrated its power.

But it was in technique and not technology alone, that Baecker helped pave the way for modern HCI. One of his other innovations was to involve trained animators in the development and testing process – well before the term "user centered design" and "participatory design" had ever been coined.

**Figure 4: In Genesys you didn't have to do frame-by-frame, or "stop" animation. You could draw the ball, and then draw the path. The path is what is shown in this image. Most important, the tablet captured the dynamics with which you drew the path, and that defined therefore determined the path and dynamics of the ball along that path.**

*Wesley Clark* is one of the fathers of the personal computer. As has already been described, he was the architect of both the TX-0 and TX-2 at Lincoln labs. He believed that "a computer should be just another piece of lab equipment." At a time when most computers were huge remote machined operated in batch mode, he advocated far more interactive access. He practiced what he preached, even though it often meant bucking current "wisdom" and authority (in a 1981 lecture, he mentioned that he had the distinction of being, "the only person to have been fired three times from MIT for insubordination.")

In 1958 the TX-0 was moved to MIT because, according to Clark, "… what MIT needed at that time was a computer that you could get your hands on. That was one thing that they did not have." That same year he helped establish biomedical computing through his collaboration on the Average Response Computer (ARC-1), which was used in experiments to average evoked responses of the brain to sensory stimuli. His next foray into biomedical computation took place in 1961 when, along with Charles Molnar, he developed the *Laboratory Instrument Computer* (LINC)[6][7]. What was interesting about this machine was that it was designed to be on a scale where a lab could have control and ownership, without the presence of the machine, by its size, expense, or complexity overwhelming the lab that it was intended to serve. What is especially telling in this regard are some of the design criteria for the machine, such as:

1. easy to program
2. easy to communicate with while in operation
3. easy to maintain
4. able to process biotechnical signals directly.
5. need be able to see over it
6. maximum cost of $25,000, the amount a lab director could spend without higher-level approval

Perhaps the best measure of their success was that the last LINC that was in regular use did not get retired until 1992. That was after 28 years of service in the Eaton-Peabody Laboratory of Auditory Physiology (EPL) of the Massachusetts Eye and Ear Infirmary.

To conclude, look again at the LINC criteria. Notice that each reflects a human-centric concern. Even today, how many systems can make that claim? Of course, the real question is, "Why not?"

*Austin Henderson* was for many years a senior researcher at Xerox PARC. He brings two things to this panel. First, as a researcher at Lincoln Labs, he did important work in interaction in his work on the implementation of the AMBIT/G graphical programming language [21]. Second, he brings to the panel a personal account of what it was like to work in this environment as a young MIT graduate student. For him the TX-2 and Lincoln Lab experience (1968-1975) was formative. It set the bar "way high" for fast prototyping of interactive applications. It was extremely important in influencing his approach to his later work in user interface management systems, like *Trillium*, as well as his work in participatory design. When one reads the literature, one repeatedly comes across the impact that people like Bert Sutherland had on them, in terms of nurturing their talent, making them feel like they belonged, and in terms of making sure that they had the support that they needed. Austin can speak to this from personal experience.

*Fontaine Richardson* led the development of an integrated circuit layout system while at the labs. The success of this work prompted him to leave the labs in 1969 and co-found a company called *Applicon*, which was one of the first CAD companies. Perhaps of most significance to the SIGCHI community was that the user interface of Richardson's system made significant use of mark, or gesture recognition, based on technology initially developed by Jim Curry at Lincoln Labs [8]. This use of gestures was carried across to the Applicon products, which meant that they were among the first commercial products to make use of marking-style interaction. Today's growth in pen-based computing, with the emergence of Tablet-PC's and electronic whiteboards, just emphasize the importance and relevance of this work to today's computational environment.

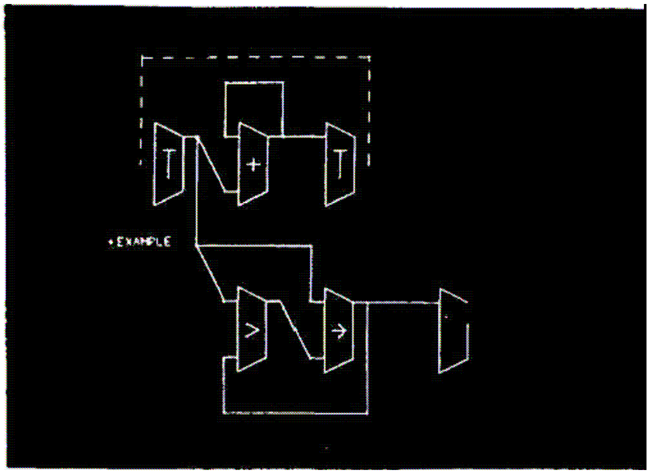

**Figure 5: A simple program defined using Bert Sutherland's graphical programming system. Each node represents a functional module, and the graphical "wires" show the flow of date from module to module. Programs were "written" using interactive graphics techniques similar to those used to make CAD drawings in Sketchpad. What was most innovative is that these drawings could execute!**

*Ivan Sutherland* is the recipient of the 1988 ACM A.M. Turing Award. In many ways Sutherland gave birth to the graphical user interface and what has become known as "direct manipulation" through his *Sketchpad* system [22]. This system was so influential that even today, one of the standard responses to a new graduate student looking for a thesis topic, is, "Go read the Sketchpad thesis [23] and pick up on one of the concepts." Sketchpad's contributions were summarized in [2] as including the following ideas and concepts:

- *Hierarchic* internal structure of computer-represented pictures, defined in terms of sub-pictures
- A *master* picture with *instances* that are transformed versions of the master; this concept helped lay the foundation for modern object-oriented programming
- Constraints as a method of specifying details of the geometry of a picture; for example, a horizontal constraint applied to a line, or an equal distance constraint applied to pairs of points
- The ability to display and manipulate *iconic* representations of constraints
- The ability to copy and instance both pictures and constraints
- Elegant techniques for picture construction using a light pen as an input device
- Separation of the coordinate system in which a picture is defined from that in which it is displayed
- Recursive operations such as "move" and "delete" applied to hierarchically defined pictures.

Some of these concepts we now just take for granted, and they are part of our everyday computational environment. Others, such as the use of constraints, are still more the exception than the rule. But taken as a whole, Sketchpad demonstrates how a holistic approach to research, oriented around a specific application domain, undertaken in the appropriate type of environment, can have a major impact. In the long run, Ivan Sutherland's approach may well exceed his technical results, in terms of importance.

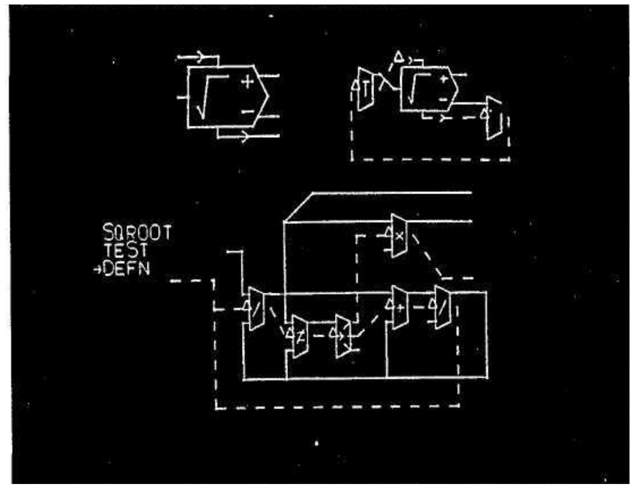


**Figure 6: An example of defining a square-root function using Bert Sutherland's graphical programing system. This is an executable program. The top left icon represents the SQRT function that is defined in the graph in the lower 2/3rds of the figure. The top right is the full program which employs the function.**

*W.R. "Bert" Sutherland* is another of the real pioneers at Lincoln Labs. His PhD thesis work [24] literally defined the notion of graphical programming. His work from the mid-60's let you interactively define a program by graphically laying out a data-flow diagram of its structure. It also then let you execute the diagram, since each graphical object represented a module, and the interconnecting "wires" defined the paths of the data flow. The system supported hierarchy, such that diagrams could be collapsed and represented by a user-defined icon. It supported iteration, conditionals, and the user could interactively change input data, or edit the program. They could even graphically define their own icons to represent modules of the program. These are techniques that even today, the vast majority of computer users, and programmers, have never experienced!

Bert then took the experience that he gained at Lincoln Labs and used it to influence three decades of research, as a research manager at BBN, Xerox PARC's Systems Science Laboratory (SSL), and finally SunLabs, (of which he was a co-founder and director).

**ACKNOWLEDGEMENTS:**

I would like to thank, Ron Baecker, Austin Henderson, Joy Mountford, William Newman, Larry Roberts, and Bert Sutherland for their suggestions and help in the organization of this panel. I mainly undertook the organization of this panel while working as a visiting researcher at Microsoft

Research, Cambridge, UK. I would especially like to thank them for their support in this endeavour.

## BIBLIOGRAPHY

1. Baecker, R.M. (1969). Picture-Driven Animation. *Proceedings of the AFIPS Spring Joint Computer Conference,* 273-288.
2. Baecker, R., Grudin, J., Buxton, W. & Greenberg, S. (1995*). Readings in Human-Computer Interaction: Toward the Year 2000.* San Francisco: Morgan Kaufmann.
3. Bell, G. (1986). Toward a History of (Personal) Workstations. *Proceedings of the ACM Conference on the History of Personal Workstations*, 1 – 17.
4. Best, Richard L. (1957). Memory Units in the Lincoln TX-2. *Proceedings of the Western Joint Computer Conference (WJCC),* 160-167.
5. Clark, W.A. (1957). The Lincoln TX-2 Computer Development. *Proceedings of the Western Joint Computer Conference (WJCC)*, 143-145.
6. Clark, W.A. (1957). The LINC Was Early and Small. *Proceedings of the ACM Conference on the History of Personal Workstations*, 133 - 155.
7. Clark, W.A., & Molnar, C.E. (1964). The LINC: A Description of the Laboratory Instrument Computer. *Ann. N. Y. A cad. Sci. 115:653-668.*
8. Curry, James E. (1969). A Tablet Input Facility for an Interactive Graphics System. . *Proceedings of the AFIPS/ ACM International Conference on Artificial Intelligence*, 33-39.
9. Feldman, Jerome A. & Rovner, Paul D. (1969). An ALGOL-Based Associative Language. *Communications of the ACM*, 12(8), 439-449.
10. Forgie, James w. (1957). The Lincoln TX-2 Input - Output System. *Proceedings of the Western Joint Computer Conference (WJCC),* 156-150.
11. Frankovich, J.M. & Peterson, H.P. (1957). A Functional Description of the TX-2 Computer. *Proceedings of the Western Joint Computer Conference (WJCC),* 146-155.
12. Frenkel, Karen A. (1989). An Interview With Ivan Sutherland. *Communications of the ACM.* 32(6), 711-718.
13. Gilmore, John T. (1958*).* MIT Lincoln Laboratory Milestones. *Communications of the ACM*, 1(8), 39-40.
14. Hurst, J., Mahoney, M.S., Taylor, N.H., Ross, D.T. & Fano, R.M. (1989). Retrospectives I: The Early Years in Computer Graphics at MIT, Lincoln Lab, and Harvard. *ACM SIGGRAPH'89 Panel Proceedings,* 19-38.
15. Hurst, J., Mahoney, M.S., Gilmore, J.T., Roberts, L.G. & Forrest, R. (1989). Retrospectives II: The Early Years in Computer Graphics at MIT, Lincoln Lab, and Harvard. *ACM SIGGRAPH'89 Panel Proceedings,* 39-73.
16. Marill, T. & Roberts, L.G. (1966). Toward a Cooperative Network of Time-Shared Computers. *Proceeding of the AFIPS Fall Joint Computer Conference***,** 425-431.
17. Olsen, Kenneth H. (1957). Transistor Circuitry in the Lincoln TX-2, *Proceedings of the Western Joint Computer Conference (WJCC), 167-171.*
18. Roberts, L. G. (1963). Machine Perception of Three-Dimensional Solids, *Technical Report No. 315*. MIT Lincoln Laboratory.
19. Roberts, L. G. (1965). Homogenous Matrix Representation and Manipulation of N-Dimensional Constructs, *MS-1505*. MIT Lincoln Laboratory, Lexington, Mass.
20. Roberts, L. G. (1966). The Lincoln Wand. *Proceedings of the AFIPS Fall Joint Computer Conference*, 223-227.
21. Rovner, Paul D. and Henderson, D. Austin (1969). On the implementation of AMBIT/G: a Graphical Programming Language. *Proceedings of the AFIPS/ ACM International Conference on Artificial Intelligence*, 9-20.
22. Sutherland, I. (1963). SKETCHPAD: A Man Machine Graphical Communication System, *Proceedings of the AFIPS Spring Joint Computer Conference***,** 329-346.
23. Sutherland, I. (1963). *Sketchpad: A Man-Machined Graphical Communication System*. PhD Thesis, MIT.
24. Sutherland, William Robert (1966). The On-Line Graphical Specification of Computer Procedures, *Ph.D. Dissertation*, Massachusetts Institute of Technology.
25. Sutherland, William Robert, Forgie, James W., and Morello, Marie V. (1969). Graphics in Time-Sharing: A Summary of the TX-2 Experience*, Proceedings of the AFIPS Spring Joint Computer Conference,* 629-636.