# Alan Kay: Transforming the Computer Into a Communication Medium

**Susan B. Barnes, Fordham University**

Alan Kay's visionary ideas about computers were instrumental in transforming the computer from an office machine into a mainstream communication device. Throughout his career at Xerox PARC, he attempted to encourage his colleagues to design small notebook size computers and create software that children could learn. Early in Kay's career, the technology was not yet available to build personal computers. As a result, Kay focused on software design and the creation of easy-to-use programs. This lead to the development of overlapping widows, graphical interfaces and the object-oriented software language called Smalltalk.

Alan Curtis Kay was born in 1940 in Springfield, Massachusetts. He was raised in an environment of art, literature, and science. At three, Kay could read and by the time he started school, he had already read several hundred books. Kay attended Bethany College, where he majored in biology and minored in mathematics. But, he was thrown out of school for protesting against the College's Jewish quota system. Kay went to Denver to teach guitar for a year and then joined the Air Force.

While in the Air Force, Kay discovered computers. He was given a two-week IBM programming course and he gained experience working with everything from a Burroughs 5000 to a Control Data 6600. Kay's concepts about programming were influenced by his Air Force experiences. After leaving the Air Force, Kay finished his undergraduate degree at the University of Colorado and went on to earn his doctorate in computer science at the University of Utah.

Utah's computer science program was headed by Dave Evans, who made all students read Ivan Sutherland's Sketchpad thesis. Sketchpad was the first interactive computer system ever build and it was the beginning of computer graphics. More important, Sketchpad provided direct visual feedback to the user.

Kay also learned about the Simula programming language developed in 1965 by Kristen Nygaard and Ole-Johan Dahl. Simula was a language used to develop computer models for production line and manufacturing systems. Simula supported discrete-event simulation construction and it introduced object-oriented programming concepts.

Kay made a biological analogy between ideas in Sketchpad and Simula. First, cells conform to basic "master" behaviors. Second, cells are autonomous and they communicate with each other by sending messages. Third, cells become different parts of the organism, depending upon the context. These ideas later became central features of the Smalltalk program design.

For a graduate project, Kay and hardware designer Edward Cheadle began designing a computer that would provide sharp graphics and windowing features. They called the machine FLEX and it was the first personal computer to directly support a graphics and simulation-oriented language. FLEX was based on previous work done by others, including Wesley Clark's LINC, a small computer that weighed several hundred pounds; the Rand Corporation's JOSS, a system designed for economists; Douglas Engelbart's interactive Augmentation System, and Seymour Papert's Logo project.

In the Fall of 1968, Kay visited Papert at M.I.T.'s Artificial Intelligence Laboratory. Papert had spent five years working with Jean Piaget, a psychologist who studied the ways in which children learn. Papert realized that the representational capabilities and responsiveness of interactive computers could be used to construct simulated "microworlds." Children could explore the computer simulated worlds in a similar way in which they explore the real world.

When Kay visited Papert, he saw children writing computer programs that generated poetry, translated English into Pig Latin, and created arithmetic environments. He then became intrigued by the analogy between print literacy and the Logo software. Before visiting Papert, Kay believed that people needed to be programmers before they could acquire computer literacy skills. After seeing children using Logo, Kay decided that computer programming languages should be developed on a level for children to understand. Children should be able to learn to read and write with this new medium.

Completed in 1969, FLEX used a tablet as a pointing device, a high-resolution display for text and animated graphics, and multiple windows. The machine also had a user interface. Kay's dissertation thesis called the "Reactive Engine" included illustrations of both complex diagrams of functions and line drawings of single-user machines. After graduation, Kay spent two years teaching at the Stanford Artificial Intelligence Laboratory, where he began thinking about book-sized computers that children could use.

In 1970, Kay made sketches of the KiddieKomp, which later developed into a laptop computer called Dynabook. The Dynabook was described as a very powerful, portable electronic device that would about the size of a notebook. It would carry an encyclopedia of information inside its circuits and plug into a computer network. Dynabook was the visionary prototype of a notebook computer that was later developed by Apple and other hardware companies.

The name, Dynabook, was influenced by the writings of Marshall McLuhan (1962,1964) who described the profound cultural impact of the Gutenberg printing press. By naming this new medium the Dynabook, Kay wanted to suggest that Dynabooks would have a similar cultural impact as the printing press. A key characteristic of the Dynabook, would be the size of the medium. Inspired by Moore's law that states the number of transistors per integrated circuit chip would double each year, Kay physically envisioned this new medium as the size of a three-ring binder with a touch-sensitive screen. Kay went to Xerox PARC to build Dynabooks.

**Xerox PARC**

In 1970, Xerox established its Palo Alto Research Center or Xerox PARC to pursue long term research to develop "the office of the future." They hired Robert Taylor, a former head of ARPA's Information Processing Techniques Office, to recruit talented computer scientists. Prior to coming to PARC, Taylor and J.C.R. Licklider published an article called "The Computer as a Communication Device,"in which they described computers as a communication medium. Licklider and Taylor realized that telecommunication networks were more than sending and receiving information from one point to another. Communicators are active participants and they play a central role in the communication process. Additionally, they argued that the digital computer is a flexible, interactive medium that can be used to support cooperative human communication. Finally, they introduced the idea of common frameworks or mental models to computer-based communication.

Taylor hired the computer scientists who built the first interactive systems, the first augmentation systems, and the first packet-switching computer networks.  In 1971, Alan Kay joined PARC and set-up the Learning Research Group.  He hired people who shared his interest in notebook size computers.  Kay has been described as PARC's "self-defined futurist-in-residence" who blurted out the unofficial PARC credo "the best way to predict the future is to *invent* it!"  The following is an excerpt from a 1971 PARC memo written by Kay:

> In the 1990s there will be millions of personal computers.  They will be the size of notebooks today, have high-resolution flat-screen reflexive displays, weigh less than ten pounds, have ten to twenty times the computing and storage capacity of an *Alto*. Let's call them *Dynabooks*. (Kay, 1996, p. 551)

While, PARC researchers developed their new computer called the Alto, Kay and his team worked on graphical user interfaces and the Smalltalk programming language.

**Graphical Interfaces**

Kay's theory for designing interfaces was primarily based on the work of Jerome Bruner. While studying children, Bruner developed a theory about different learning mentalities.  The first stage Bruner (1966) called *enactive*, or learning through action.    The second stage, *iconic*, uses a "system of representation that depends on visual or other sensory organization and upon the use of summarizing images" (pp. 10-11).   Finally, in the third stage, *symbolic*, representation in words or language occurs.

Kay constructed a model called "Doing with Images makes Symbols."

| | | | | |
|---|---|---|---|---|
| DOING<br>with | mouse | enactive | Know where you are, | |
| IMAGES | icons,<br>windows | iconic | recognize, compare,<br>configure, concrete | |
| makes<br>SYMBOLS | Smalltalk<br>language] | symbolic | tie together long<br>abstract | [computer |

(Kay, 1996, p. 551)

A variation of Engelbart's mouse would be used as a form of enactive representation to actively navigate and manipulate text and icons displayed on a computer screen.  Icons and windows were incorporated into the design as a level of iconic representation. The Smalltalk programming language was the symbolic level of the design

**Smalltalk**

By 1973, Kay's Learning Research Group had eight members, including Dan Ingalls, Diana Merry, Ted Kaehler, Adele Goldberg, Larry Tesler, and Chris Jeffers. Ingalls took on the primary role of transforming many of Kay's ideas into reality and Goldberg was an educational specialist.

Originally, Kay thought that Smalltalk would be an iconic programming language and it would take two years to develop. However, after a hallway discussion with Ted Kaehler and Dan Ingalls, Kay was challenged to define a new computer language on a single page. For two weeks, Kay worked on the problem from four in the morning until eight at night. The result was the outline for Smalltalk, which was then implemented by Dan Ingalls. Smalltalk was selected as an innocuous name for the program to contrast other programming systems that were named after Greek gods, such as Zeus and Odin. Over the next ten years, Kay and his team made over 80 variations of Smalltalk which features visual feedback and accessibility to novice users.

The Smalltalk programming language was developed first and it was then used to build an operating system and a graphical user interface. On top of the Smalltalk language and the graphical user interface, the Xerox researchers built an entire programming environment consisting of editors, debuggers, compilers and then used those to carry out several large applications such as music and animation systems.

Early versions of Smalltalk were tested with local school children. Kay (1996) believed that "the content of personal computing was interactive tools, the content of this new kind of authoring literacy should be the creation of interactive tools by children" (p. 544). The children were taught programming by working from examples of serious programming problems. For example, the children were taught how to animate a simple box. Soon the children were creating paint, music, illustration, and animation tools. Over four years, Kay and his team invited over 250 children aged six to fifteen and 50 adults to try versions of Smalltalk with its interface. Kay then used the visitor's suggestions and the projects created with the system to improve the design.  These experimental projects included:  programs for home accounts, information storage and retrieval, teaching, drawing, painting, music synthesis, writing and games.   But, they encountered the following problem: in what order and depth should programming be taught to children?  Their early successes could not be extended to larger groups of children.

Smalltalk ran on an Alto, which was designed as a desktop computer with a specially constructed monitor.  In contrast to typical computer terminals that display a standard set off characters, documents displayed on the Alto's screen looked like they were typeset.  Text clarity and readability was an important Dynabook feature, which the Alto designers helped to develop.

The Alto's 81/2-by-11-inch screen was limited in terms of the amount of information it could display.  The PARC researchers began to think about the screen in terms of a physical desktop.  Kay realized, that people pile pages on top of each other on a desk.  Similarly, they could place windows on top of each other on the screen.  From this analogy the idea of overlapping windows was developed.

In the Fall of 1974 Ingalls created a program called BitBlit, short for bit boundary block transfer, as a way to shift rectangles on the Alto's bitmapped display from one location to another in a single operation.  BitBlit enabled one window to be hidden behind another and it created the illusion of different layers.  Kay and his team worked BitBlit operations into the user interface design.  Different programs where placed in multiple windows, which were effortlessly shuffled on the screen like pieces of paper.  Thus, the metaphorical desktop was created and it became a mental model for future interface design.  The Alto was the first computer to use a mouse with the popular desktop environment of icons, folders, and documents, which is used today by Macintosh and Windows.

**Notetaker**

By 1976, Ingalls and Kay's team at PARC had transformed Smalltalk into a full-service programming language.  Smalltalk—76 was two hundred times faster than Smalltalk—72.  Kay shifted his attention toward a new project called Notetaker, another notebook sized computer. The central idea was to take a percentage of the Alto's functionality and put in a compact portable machine.   The Notetaker design included a custom-built display screen that was touch-sensitive to eliminate the mouse; stereo audio speakers with a built-in microphone; 128,000 bytes of main memory, a rechargeable battery; and an Ethernet port. However, the technology at the time was highly limited.

The actual Notetaker was a plump attache case that looked like the first generation of "luggable" computers that were created six years later. The case flipped open with the screen and disk drive set in the larger half, which faced the user when the box was laid flat on a table. The smaller section contained the keyboard that connected to the larger half by a flexible cable. It was difficult to carry, so Larry Tesler and Douglas Fairbairn built a rolling cart for it, which allowed them to slide the computer under an airline seat. Fairbairn became the first person to use a personal computer on an airplane when he turned on the battery-powered Notetaker during a flight to Rochester.

Ten Notetakers were built and team members flew around the United States trying to get Xerox managers interested in the product. But, they never produced it. In 1980, Kay took a sabbatical from PARC leaving Adele Goldberg in charge of the group. She led the team in the development of SmallTalk—80, which was later marketed by a Xerox offshoot called ParcPlace. For almost thirty years, Smalltalk has been the language of choice for high-complexity applications, such as Texas Instrument's semiconductor manufacturing system.

**After PARC**

Kay left PARC in the early 1980s to move to Los Angeles and take organ lessons. In 1983, he spent a year at Atari before joining Apple Computer. At Apple, Kay put together a team, including Dan Ingalls, Ted Kaehler, John Maloney, and Scott Wallace to develop Squeak. Squeak, is an object-oriented educational program based on Smalltalk, but it has additional multimedia features. Launched in 1996, Squeak is an open-source-code language that is available through the Internet. The purpose of the Squeak project is to create an education platform that could be programmed by children and nontechies.

In 1997, Kay moved his team to Disney's Imagineering Division and they continued to work on the Squeak project. Presently, Kay is setting-up Squeak Land, a nonprofit educational web site to create and distribute Squeak programmed multimedia documents. He continues to grapple with the issue of teaching children how to use computers. Kay's largest contribution to computer science is the influence he has had on shifting the computer paradigm away from a text-based office information system toward a personal multimedia communication device.

**References and Further Reading**

Bruner, J. S. (1966). *Towards a theory of instruction*. Cambridge, MA: Harvard University Press.

Hiltzik, M.A. (1999). *Dealers of lighting: Xerox PARC and the dawn of the computer age*. New York: HarperCollins.

Kay, Alan C. (1991, September). Computers, networks and education. *Scientific American*, pp. 138-148.

Kay, Alan C. (1990). User interface: A personal view. In Brenda Laurel (Ed.), *The art of human-computer interface design*, pp. 191-207. Reading, MA: Addison-Wesley Publishing.

Kay, Alan C. (1977, September). Microelectronics and the personal computer. *Scientific American*, pp. 231-244.

Kay, A. (1996). The early history of Smalltalk. In T.J. Bergin & R.G. Gibson, eds., *Programming languages*. New York: ACM Press.

Licklider, J.C.R. & Taylor, R. (1968, April). The computer as a communication device. *Science and Tedhnology*, [Online] 41pp. Available: http://gatekeeper.dec.com/pub/DEC/SRC/research-reports/abstracts/src-rr-061.html. [July 11, 2001].

McLuhan, M. (1964). *Understanding media: The extensions of man*. New York: Signet Books.

McLuhan, M. (1962). *The Gutenberg galaxy*. Toronto: University of Toronto Press.

Rheingold, H. (1985). *Tools for thought*. New York: Simon & Schuster.

Shasha, D. & Lazere, C. (1995). *Out of their minds*. New York: Copernucus, Springer-Verlag.

**Web Sites**

www.squeakland.org
squeak.cs.uiuc.edu