Oral History of Andries “Andy” van Dam

Interviewed by:
Marc Weber

Recorded: December 10, 2008
Menlo Park, California

CHM Reference number: X5675.2010

Marc Weber: I'm Marc Weber of the Computer History Museum, and I'm here with graphics and hypertext pioneer, Andy van Dam. Today is December 10th, 2008. First, I want to thank you for sitting down with me and talking about your work.

Andries “Andy” van Dam: Sure, my pleasure.

Weber: I just want really to start at the beginning and give a little bit of background on where you grew up and what got you interested in computing.

van Dam: How far back do you want to go in the question of where did I grow up? I emigrated from the Netherlands, where I was born, in 1952. I had three years of schooling in Cape Cod. I did my college and graduate work in Pennsylvania and then moved back to New England, and have held only one job since, with a couple of exceptions for sabbaticals. That’s been at Brown University since 1965.

Weber: You went to Swarthmore?

van Dam: I went to Swarthmore as an undergraduate, where I met Ted Nelson. In fact, freshman week.

Weber: You played Frisbee together, I believe?

van Dam: Frisbee, we had spirited arguments, we had a shared interest in theatre. He produced the world’s first rock musical. My wife—who at the time was my girlfriend—and I had bit parts in that show. Then I reconnected with Ted after we were—in my case, I was out of grad school and already teaching, and he was working on his hypertext ideas. They later crystallized in Xanadu.

Weber: You were familiar from the beginning with some of those ideas from meeting him.

van Dam: Well, from the beginning for me was 1967, when I met him at a computer conference and we saw each other and decided to catch up. I learned about this idea that he had. At that time, I had taken possession of a real treasure, which was an IBM [International Business Machines] 2250 display, which we had gotten because of the close connection between Brown University and [IBM President and CEO] T.J. Watson, who graduated from Brown. We were very fortunate to be able to get one of those very expensive displays. Even though the display was for doing research and graphics, Ted convinced me that there was an exciting possibility of looking at hypertext and how that might be implemented. I did a bootleg project which became the Hypertext Editing System.

Weber: When you knew him at Swarthmore, he may have had ideas, but he didn’t discuss them.

van Dam: No. I had no knowledge of these ideas of his about how information should be organized, stored, retrieved, and so on.

Weber: Had you read the Bush article, “As We May Think”?

van Dam: No, I had not. No, that didn’t come until later.

Weber: Graphics; describe your early interest and what got you into computer graphics.

van Dam: What got me hooked effectively was a life-changing experience for me; it was watching the Sketchpad movie. I saw that and had an epiphany and said, “Okay, that’s what I’ve got to do with the rest of my life.” Now, pictures weren’t exactly foreign to me. I had worked on character recognition already in 1960, fresh out of college, as part of a summer job at Burroughs Research Center in Paoli, Pennsylvania. I worked with the great C.K. Chow as his apprentice. He was one of the world’s experts in character recognition. I had also been interested in how one could represent simple two-dimensional images. I had done a master’s thesis on a blueprint repository using aperture cards. Aperture cards are IBM cards, as they became known, and are 80-column punched cards with a slot in them where you inserted a piece of microfilm so that you could store digital information along with the analog image. There was a huge repository of blueprints in Mechanicsburg, Pennsylvania, among other places. It’s how the Navy kept all their documentation and made it machine searchable. I did a master’s thesis on how you could do Boolean keyword retrieval on aperture cards.

Weber: When you say digital...

van Dam: They had punched columns, so you could have alpha-numeric information. But these in particular stored with the microfilm.

Weber: But then the microfilm of course would be an image, so you would—

van Dam: Microfilm was an image. I think the columns of digital information that you had available were purely for identifying the image and maybe having a few simple keywords.

Weber: Metadata.

van Dam: Metadata is what we would call it today, exactly right. I had always been interested in things visual. I did the usual high school photography, was yearbook photographer. Imagery to me was a cool thing. When I saw somebody manipulating images on the screen in real-time in a world that for me at the time, and for everybody else, consisted of feeding decks of punched cards to computers, that was just mind-blowing.

Weber: Where were you when you saw that?

van Dam: I was a graduate student at the university. I don’t even remember how I came to see the movie. But as I said, it changed my life. I decided I had to do something about manipulating imagery. Unfortunately, we didn’t have a display. We barely had a computer that we had access to. In fact, sidebar, my first programming course, we didn’t even get our hands on a computer, everything was hand checked. Computers were hard to come by in the early 1960s. To loop back to the event in 1967, I had then for the first time a real live interactive display. I had written about interactive displays and other imaging technology in my thesis and I had worked on data structures for representing two-dimensional images, but I hadn’t actually been able to do it all real-time on the display. We simulated output on the line printer, and it was amazingly hokey compared to what we have access to today, but those were the early days.

Anyhow, being able to get my hands on a tube and have it in my lab and accessible 24 hours a day, if I could get the machine time, which was not so easy, that was a real thrill.

Weber: That was a very expensive display.

van Dam: Yes. I think it was on the order of a quarter of a million dollars, which by today’s standards would be several million dollars. 1967 money.

Weber: Ten times inflation roughly.

van Dam: Yes, something like that. The mainframe, which at that time was 512KB [memory capacity], was probably a million dollars or more. Essentially, we had to sign up for time, typically third shift, to be able to run the tube. Then after a while, we were able to run it in a partition of a multi-programmed system, a multi-programmed operating system. Eventually timesharing became sort of the standard way of hooking peripherals.

Weber: When you had the epiphany with the movie, did you have to change what you were doing completely?

van Dam: I hadn’t been entirely sure whether I would continue with information retrieval or do something quite different. It came at a moment when I should have made a decision anyhow. The decision came to try to understand how you represent pictorial information. That’s what the Sketchpad thesis was about to a large extent, but I thought there were other things that could be done, and I used a different approach than Ivan had used for Sketchpad.

Weber: What were some of those other things?

van Dam: Well, that’s actually an interesting part of the story. I used what was called an associative memory system or simulator, called Multi-list, which was a simple but effective simulation of associative memory of the type that it was thought we had in our brains, à la Bush, associative trails. What this allowed you to do is to have lists, a list per keyword. When you tagged an object of some kind with say, half a dozen keywords, it would appear on the keyword lists for each keyword. Then in the simulator, you would be able essentially to tug on the threads that represented those lists and get the intersection of all of those terms. What would come back would be the item that appeared on all six lists if you specified a keyword retrieval of six items, six keywords.

Weber: A common denominator of association in a sense.

van Dam: Yes. Effectively it was an “and” of those six keywords. I won’t claim to have any, let’s say any responsibility for the associative memory simulator. I didn’t implement that, I was a user of it. I figured I could use that kind of representation to store images. That’s what my thesis was about mostly.

Weber: Purely image to image with no words.

van Dam: Yes, exactly. Two-dimensional line drawings. It didn’t deal with 3-D. But I did deal, for example, with how you transform sub-images to create compositions of images or objects, really. In the same way that Sketchpad did that. That was the major contribution of Sketchpad, if you ignore the hardware. He designed these data structures that let you have hierarchies of components. Hierarchies and sub-hierarchies sort of indefinitely down. So I found a different way of representing them using this Multi-list idea.

Weber: But graphics primitives and then Sketchpad model.

van Dam: Well, graphics primitives, and you combine primitives to make what we would call graphic objects today. You sort of built from the bottom up and make ever more complicated assemblies.

Weber: Which was your approach.

van Dam: The key Sketchpad idea. I had a different way of representing things, but in many ways it duplicated the key ideas of Sketchpad.

Weber: This was your doctorate work then.

van Dam: Yes. I also described a number of technologies that I thought would be important for image storage and retrieval, including such things as plasma displays, which I had just been learning about and which I was given to understand would probably arrive on the scene within five to ten years. Took a lot longer before we had plasma displays.

Weber: Where did you see them?

van Dam: I forget where I learned about them. I made a lot of phone calls and did a lot of social engineering to get state of the art information about what people were working on. That was another very descriptive part of my thesis.

Weber: You were not—Computer science as an academic department didn’t exist.

van Dam: Well, actually yes I was. Penn was arguably the first school to have a PhD track in what was called computer and information sciences. My good friend to this day, Dick Wexelblat and I, and a number of other people, were the first enrollees in that PhD program. Dick got his PhD in 1965, I left in 1965 to go to Brown, and finished up my dissertation while at Brown by commuting to Penn and getting machine time at three o’clock in the morning.

Weber: Commuting, literally? Wow.

van Dam: Yes. Kind of commuting. I finished up the degree in 1966. He and I have the first two PhDs in a doctoral program that specifically was identified with computing. There were earlier PhDs, of course, numerical analysis and various other precursors to modern computer science. But they weren’t given by programs specifically called computer anything.

Weber: Similar to a computer science program today, or very different?

van Dam: Oh, you could see that it was embryonic. In particular, when I got to Brown I found that most of the material that I had learned I couldn’t really use, except for some of the fundamental logical design and switching theory. The stuff I learned about languages and translation techniques really wasn’t state of the art anymore. My first few years, I had to learn a lot of new stuff so that I could then teach it. I worked a huge amount with undergraduates who were learning this material with me. Then we would sort of teach each other, and then we would give it to the class. The classes in those days got research results almost from the get-go.

Weber: Why did you go to Brown?

van Dam: That’s a circuitous story in and of itself. You want me to tell the story? My wife, before we had twins, was a high school teacher in the local high school. She taught French and belonged to the NEA, National Education Association. They have a magazine. I read that magazine, just leafing through it. There was an article about a guy named George Grossman, who in the New York City school system—it might have been Bronx High School of Science, but I don’t remember that—was teaching his high school kids about computing and teaching them to program. At first, I, being a thorough skeptic then and still, said, “Oh, come on, no way. I’m learning this as a graduate student. How can you possibly teach that stuff to high school kids?” But after the first reaction went away and I started actually thinking about whether it was feasible, I came to the conclusion that well, of course you could do that. It didn’t involve a lot of physics or higher mathematics. It was logical and logic, and there really wasn’t any reason that you couldn’t teach it to high school students. That led me to the idea of maybe offering a few hours of instruction to a few of Debbie’s brightest students. That seemed like a lot of work for very little return. So the scope and ambition kept growing. To cut to the chase, I mounted a full summer’s program for high school students and high school teachers together. That involved both some suburban schools, including Debbie’s, and some schools in the Philadelphia school district. There were about 30 high school teachers and students. We published the results. It was a gas, just the most fun I’d had in a long time. It changed my mind about what my career path should be. I had no intention of going into academia. My father had taught college, taught biology to premeds, and so I knew I wasn’t going to do biology and I wasn’t going to go into teaching. But I had so much fun teaching this course in 1962 that I decided that I should go to academe. Okay. So I’m close to finishing my dissertation, or not so close as it turns out, in the fall of 1965. I get a call from a graduate of that program, Jim Castellan. He calls me up and he says, “Hey, did you know there is an opening at Brown?” I said, “Well, no I didn’t. But I’ve all but accepted a job at University of Maryland, so it’s kind of too late.”

Weber: To do? Maryland would have been—

Van Dam: To teach. To teach computer science. He said, “Oh, you’ve got to come to Brown.” I said, “Why?” He said, “Because Brown really believes in undergraduates and undergraduate teaching.” He knew I’d had a wonderful time at Swarthmore and had really appreciated the intimate contact with faculty and small classes and the whole liberal arts college experience. He knew I wasn’t very happy as a graduate student at Penn because it was large and impersonal and it was a commuting school, and a lot of reasons. He said, “You would really fit well at Brown.” I decided, “All right, what the hell, I’ll invest one day.” So I spent a day. I was met by one of the senior faculty members at the airport who picked me up. I interviewed with the chairman of the department, who excused himself in the middle of the interview to go teach a freshman course. I said, “Ah ha! They mean it.” You, being a Brown graduate, know that Brown

takes undergraduate teaching very seriously. At Penn at that time—this is a very different Penn from the Penn of today—you could tell the rank of a faculty member by what level course they taught. Assistant professors did not have contact with freshman necessarily, but certainly, senior faculty didn’t. They taught the advanced courses.

Weber: Maryland would have been similar?

van Dam: Maryland, my guess is, would have been similar. But here, the department chairman taught freshman. That would never have happened at Penn.

Weber: Even at Swarthmore.

van Dam: Oh yes.

Weber: Yes?

van Dam: Oh, absolutely. At Swarthmore, the departments were sufficiently small then. Everybody taught, nobody was just an administrator.

Weber: What was the department at Brown?

van Dam: It was applied math. Good question. The other thing that convinced me to go to Brown is they had this wonderful [Arshile] Gorky tapestry in their computing lab. The computing lab was just new then. I thought, wow, anyplace that hangs a Gorky in a computing lab, that’s a touch of class I really liked. Plus, I love New England, having grown up at least on this side of the ocean in Woods Hole and Falmouth. There were a number of reasons that made me go to Brown. Fortunately, I made the right decision and I’ve been there ever since. That’s kind of a funky chain of coincidences that got me into teaching and then got me through that teaching in particular to go to Brown, where Jim was my student and his wife to be was my student and my teaching assistant. So it’s a fun story.

Weber: Your wife continued doing education?

van Dam: No. After the twins were born, she stopped being an educator.

Weber: The position at Brown was for computer science.

van Dam: They hired me knowing that I was a computer scientist, not an applied mathematician, and that I wanted to set up a computer science department. It took a lot longer than I thought it would. Eventually, some folks in engineering and some folks in applied math got together and over a period of years, convinced the administration that they should let their people go.

Weber: But the person that hired you knew that they were potentially planting that seed.

van Dam: He hired me because they wanted to have a real computer scientist. Brown was one of the very first schools to teach computer science to undergraduates. By 1966, there were several tracks in applied math. There was the traditional applied math track, and then there were people who were working with me and subsequently other faculty were hired who took some applied math, but their courses were primarily in computer science. This was at a time when all the top schools, the ones with good graduate departments, said, “No, no, no, that’s a premature specialization. An undergraduate may take a programming course, but we’re certainly not going to give lots of specialized courses at the undergraduate level.” They didn’t move from that point of view until the 1980s or 1990s, for some of the top schools. Brown had been doing it since the late 1960s. That accounts, by the way, for the fact that there are a very large number of influential folks in the computer industry who started out when they were undergraduates, did research with me, were teaching assistants with me, and have really make an indelible mark on the industry. Brown specialized in undergraduates since 1966, I can say.

Weber: Brown was the first or second?

van Dam: To do undergraduates? You know, I’ve not tracked that history. There might have been another place, but I certainly can’t come up with one. CMU [Carnegie Mellon University] did not, Stanford did not, MIT [Massachusetts Institute of Technology] did not, Cornell did not, Berkeley did not. They started their undergraduate programs much later. They started with graduate programs that had been well underway by the time that I started the undergraduate program.

Weber: So you finished your PhD, that was on the graphics, the associative model.

van Dam: Yes. Using the associative memory simulation to do two-D graphics.

Weber: But doing all of this without an actual display.

van Dam: That’s right. Using a line printer for output. There was no interactive input, so getting the 2250 was growing up and moving from the tricycle era to a motorbike in one fell swoop.

Weber: You had twins at this point.

van Dam: Yes. One of the good family stories is that the weekend we arrived, my wife and I were typing our dissertation on typewriters with the twins lying on the couch all through the night. That was the first of many all nighters all through the year. So yes, that was a bit hairy that first year especially, and the kids were small, and I was commuting to Philadelphia, and establishing brand new courses that I’d never taught before and for which I still had to learn the material. Get the research group started, helping to lobby for the 2250, and getting people to start working with it, and so on. But it was fun. It was really great fun. The year after that, I went to see Doug Engelbart’s show.

Weber: But 1967 is when...

van Dam: When I met Ted Nelson.

Weber: You met Ted again.

van Dam: When some undergraduates and I started working with Ted as advisor, consultant, requirement setter, inspirer to—

Weber: Where was Ted at the time?

van Dam: Doug was at the—

Weber: No, I’m sorry. Where wasTed when you met him?

Van Dam: Ted, I think, was living with his grandfather still. He was commuting. We didn’t have any money to pay him, so he did this—

Weber: But from New Jersey?

van Dam: From New Jersey, probably. Yes, or it might have been from New York. I really don’t remember.

Weber: You told me you met him at—

van Dam: I met him at the Spring Joint Computer Conference. In those days, Spring was typically on the East Coast and Fall Joint Computer Conference was on the West Coast.

Weber: How was he regarded by the computer scientists?

van Dam: They thought he was pretty weird, pretty far out there. He had all these wild and crazy ideas. The collaboration was not always easy. As you know, Ted was ultimately very disappointed in what we produced because it didn’t resemble the Xanadu vision in any way that he would recognize. For us, it was our first pancake, and we couldn’t have done anything more ambitious with the resources that we had and the knowledge that we had. I still think it was a very good first effort. But we threw out most of the design and the ideas for the second system that we implemented without Ted’s input.

Weber: The first was HES.

van Dam: First was the Hypertext Editing System, HES. The paper, Carmody, who was an undergraduate at the time, and Ted and me as the authors, describes it quite well. Then came FRES, File Retrieval and Editing System. It’s a pun; we thought the system was a “fresser” in Yiddish in that it consumed, oh my God, 128KB of memory in one of the partitions of this multitasking operating system. That was an immense amount of memory. It was a 512KB machine. To use up 128KB partition, that was a memory hog, hence the pun.

Weber: So Ted was coming, you were developing this with his—

van Dam: During the Hypertext Editing System days, right.

Weber: At that point, you knew nothing about what was going on at SRI [Stanford Research Institute]?

van Dam: I had no knowledge of SRI. Ted had known of Engelbart, I believe, but we certainly didn’t discuss it. I’m not so sure when he learned about what Engelbart was up to. I don’t even remember when we read the Bush paper for the first time. The first system was completely inspired by Ted’s ideas. We tried to cut them down to size to the point where we thought we could implement them, which is the source of the profound disappointment that Ted has experienced ever since, that we realized so little of his vision. As I said earlier, I’m still proud of what we got done in a very small amount of time. By the time I saw the mother of all demos in 1968, probably more than anyone else in the audience, I appreciated the breathtaking breadth of Doug’s vision and what he and his group had pulled off. As Doug himself says on one of the web pages that describes this, I was very skeptical, and believed that some of it must have been demo-ware, the man behind the curtain, etc. He was very, very kind and said, “No, come up to my lab and we’ll show you how it’s done.” They “opened the kimono,” as it was called many years later in Silicon Valley, and essentially gave me access to all of the technology and how they had done it. I wound up writing up the system aspects, the algorithms and data structures that they used to do much of the manipulations that he demoed in 1968. I did that as part of a paper that compared and contrasted their way of doing it with lots of other ways, including our own in both HES and FRES.

Weber: What were the key differences?

Van Dam: The key differences were that he was very hierarchy oriented. Everything in NLS [oN-Line System] is a statement. Those statements had 4,000-character limits, which were not unreasonable, given the severe memory constraints at the time. But Ted had always insisted that we weren’t going to have lines and we weren’t going to have statements. There were some program editors that had like 512-character strings. He wanted completely freeform and arbitrary-length strings. I agreed that that was a good way to think. I didn’t even really think about hierarchy until I saw NLS. So in FRES, what we did is to combine Ted’s idea of arbitrary-length strings with optional hierarchy that you could superimpose on the strings. We copied the view specs idea, and we had keyworded access down to the character level if you wanted that. But we preserved a lot of the freer nature of the Hypertext Editing System. We also made links bidirectional, and kept them fine-grained.

Weber: NLS links were bidirectional, though, no?

van Dam: I don’t recall whether the NLS links were bidirectional. You could certainly go back. We had that also in the Hypertext Editing System. But there’s a difference between bidirectional links, where you can always see who’s pointing to you, and being able to reverse the trail.

Weber: You had a database of links?

van Dam: We did not have a separate database of links in those systems. That is one of the things that Norm Meyrowitz, who designed the intermediate system as its principal architect, developed as a novel idea, that you separate out the link structure from the raw content so that you can have multiple link structures over the same content. None of NLS, HES, or FRES would have been able to do that.

Weber: So if a link was broken there was no obvious check except to try it then.

van Dam: Well, I guess the only way a link would be broken is if the destination would somehow have been deleted. Yes, that would give you an error. I believe that in FRES, if you touched a piece of text that was being linked to, we would go back and essentially make the link pointing to that, or any links pointing to that invalid. Because you had the information it was pointing to you. If the anchor for the link went away, then we could go back into the page structure and delete the forward pointers to that point. I don't think it was possible to get a 404.

Weber: Tell about how this fit in with your graphics work.

van Dam: Oh, well, it didn’t, is the answer. HES was a complete bootleg project. It was done in spare time by these undergraduates who were having a ball learning how to do this kind of stuff.

Weber: Who were the main people?

van Dam: Steve Carmody, and he is still at Brown. He does various kinds of sort of technology work in the computing center, at one point was the head of the computing center. I don’t remember now who else worked on HES. I have a better memory of who all worked on FRES, which was quite a sizeable group, including Bob Wallace, who was one of the co-founders of Microsoft, and the inventor of Shareware, and died, unfortunately, prematurely. HES was a bootleg project, but eventually I showed it to our program monitor, Sam Matsa, and convinced Sam that this kind of use of an expensive graphics terminal could lead to something useful that we could perhaps influence “Time Magazine,” and the “New York Times,” to think about having their editors enter their stories online and edit them, and see the changes in real-time instead of phoning them in. Remember, this is before even acoustic couplers, and editors did not sit at terminals in those days. We demoed at the Time-Life Building once Sam and IBM became reconciled to the fact that we’d sort of bootlegged this little project, they were actually were excited about it. For about six months, Sam and his right-hand man, Don Ruley [ph?] at the IBM New York Scientific Center, which I think was on Fifth Avenue but I’m not entirely sure. Ted and I on our end were going to go into business together to create a version of Hypertext Editing System/FRES/Xanadu that we would try to flog commercially. We went as far as to talk to a venture capitalist. In the words of the old limerick, we argued all night as to who had the right to what with which and to whom. Eventually, the venture fell apart, and Ted and I went our own separate ways. But, to hook back to another part of the story, Sam at that time was very involved in ACM [Association for Computing Machinery], and thought at some point he might like to be the chairman or the director of ACM. He was very involved with the idea of having ACM put on professional development seminars. He and Jim Adams, who was a full-time employee of ACM, and I agreed that one of the very first should be on computer graphics. Sam and I put on a set of events in various cities around the country in 1967 and 1968 to bring graphics to people who were interested in getting a quick one-day introduction. I did most of the lecturing, Sam had one piece. That, in turn, led us to see that there was a lot of interest in graphics. We sometimes had as many as 80 to 100 people in a ballroom. We then said, “Gee, you know, there seems to be some growing interest in computer graphics, it’s not this narrow niche specialty anymore practiced by a few people in aerospace and automobile companies. Maybe it’s time to start a little professional activity.” That became SICGRAPH, Special—

<crew talk>

Weber: If you could say again the origins of SIGGRAPH [Special Interest Group on Graphics and Interactive Techniques].

van Dam: Yes. Seeing that there was significant interest because of the professional development seminars, we then decided that we should have a little professional group within ACM. Sam was the chairman and I was the secretary. Before we could even get that off the ground, the ACM bylaws required that you be able to show that this was a field that had some promise by getting a mere 30 signatures. It was tough getting 30 signatures on that petition. There weren’t that many practitioners. There were people who were interested, but the number of people who were doing work in the area was tiny. This was a complete niche field when we started. We eventually got them. This was a very low-key activity. We produced some newsletters at Brown using the Hypertext Editing System, which I can’t find, unfortunately. Then nothing much happened until 1974, when some other people established a conference, and that conference eventually became this monster event called SIGGRAPH, the SIGGRAPH Annual Conference, with all of its associated activities, its newsletters and publications and educational activities and exhibition, which at one point had all of the giants in the industry. A huge amount of space with exhibits and even today, with a shrinking field and economy, there are still 25 - 30,000 people who come every year. It’s by far the largest group under ACM sponsorship. Now, most of those people are not doing research. The number of graphics researchers, or people who are interested in graphics research is probably five, six, seven thousand. They go to the technical program. Most people come from the exhibition and the art show and so on. So there’s a little bit of a disconnect between what Sam and I formed in the late 1960s and the formation of SIGGRAPH as we know it today. Other people get the credit for that.

Weber: But when you say “nothing much happened between 1968 and 1974,” you continued to meet.

van Dam: Not really.

Weber: No.

van Dam: No. It was more a virtual organization. There was a chapter in LA. [Los Angeles], because there was a lot of aerospace industry in LA. But no, I would say it was fairly virtual as an organization.

Weber: But it continued to exist.

van Dam: It continued to exist, certainly legally, in terms of the ACM bylaws. And at some point, it was converted to be SIGGRAPH with a G as in group.

Weber: Why was it SIC before?

van Dam: SIC, I think you started as a committee and then you became a group when your activities warranted it. Unfortunately, that’s a bit of the history that I don’t have any more. Sam might, Jim Adams might. I don’t know. To come back to another thread, which is the trying to convince people that someday we would all sit behind screens and—

<audio ends abruptly>

END OF TAPE 1

START OF TAPE 2

Weber: You were circling back from SIGGRAPH.

Van Dam: Yes, right. That was the SIGGRAPH story. Now back to the hypertext story. Through Sam Matsa's connection at *Time-Life* and the *New York Times*, we exhibited. We did a demo in the Time-Life building. We had people from the *New York Times* come to Brown, and Ted had constructed a very nice web of electroplating patents, and had done essentially the information architecture. That was our standard demo.

Weber: Electroplating—

Van Dam: Patents, yes. There are cross-references, so there was a sort of hierarchical way of coming down, or you could be in any one and see links to others. For the time, it was quite an elaborate little web. He had very carefully thought about ways through it, and we knew how to demo it. The reaction, I remember well enough as well. "This is really interesting, but it's too futuristic for us, editors in front of expensive screens. Not so likely." I believe *Newsday* in Long Island had editors behind ATEX timesharing system screens in '76. Less than a decade after we were told we were way too far into the future. What had to happen is, that the cost had to come down. We were still running on a very expensive display, connected to a very expensive mainframe. What the ATEX people did was to use a minicomputer and use very cheap displays. They were very much less powerful than what we had in terms of their functional capability, but correspondingly, also easier to understand. The hypertext editing system, we had a function keypad with 32 keys, and overlays, which could be sensed by the computer. We had effectively two sets of functions. One was for editing, one was for formatting. I learned very quickly that when I had an overlay labeled with 31 functions, people's eyes would glaze over. So I made overlays where the buttons were hidden, except for the middle row, which was just insert, delete, move, delete and maybe one more. I would show that with the light pen, how you indicated a text string, and then push the button to have something done.

Weber: But the light pen was the pointer.

Van Dam: The light pen was the mouse equivalent. That was the pointer, exactly. After they mastered that, then I would say, effectively, "But wait, there's more!" and then bring up the next overlay, which would expose another set of functions.

Weber: How would they add a link, for instance?

Van Dam: That was an insert link, effectively.

Weber: Then they would—

Van Dam: And then you penned the explainer on the link.

Weber: So typed links.

Van Dam: No, you didn't type them in. You had to make them by specifying source and destination. By the way, that was one of the good things that Doug’s team invented, certainly in the hypertext editing system in the FRES. You couldn't type your links in. You had to indicate them by direct manipulation. You had to show the source and the destination, and then specify either before or after—I've forgotten exactly—I think probably in prefix mode. So first the command, and then the source and destination. Then we would fix up the data structure to record that link. In Doug's case, you could actually type the names of the statements, the labels of statements, and that would establish the link with a special command. It was a better design, because you could edit the link as if it were normal text. You couldn't do that in our systems.

Weber: His links, they were not all simple association. It could be defined as like, this link is to a definition, or this link is to—

Van Dam: Well, at some point, you could add view specs to links. I don't know whether that was present in the '68 demo already. In FRES, we had view specs with links, we had keywords with links, so you could be quite fine grained about your control of what a particular user of the system would see. We had other kinds of links that were completely hidden, so you could splice things together, without having to explicitly link to go from one place to another. We distinguished between jumps and links—sorry, jumps and splices.

Weber: A splice was hidden. How would it manifest itself?

Van Dam: You would simply move from one to the other smoothly. There must have been a view spec in which you could actually see the splices. That would have been a sort of editor's view spec. The reader would have jumped seamlessly from one to the next, so that you could splice together.

Weber: To what extent was there text editing on computers at this time?

Van Dam: Oh, there wasn't.

Weber: None.

Van Dam: Oh, no.

Weber: Is it accurate that both you and Doug essentially independently—

Van Dam: We had the first word processors, in effect. What people had was line editors, or Lampson's QED [quick editor], which was a more powerful line editor, because it wasn't just restricted to single lines. I think he went up to 512 characters. I wrote that all up in several papers. In any case, it's clear that people used text editors, program text editors, for producing documentation. That was an obvious thing to do. Our systems had way more functionality for editing, and they had hypertext linking. From the beginning, the hypertext editing system also was able to accept formatting codes, and used IBM's Text 360 program to do nice printing on the line printer. You could specify paragraphs and tabs and indents, and stuff like that, if you had the function keypad.

Weber: How many lines could you see on the screen, roughly?

Van Dam: Oh, quite a few lines. Certainly 24.

Weber: It's accurate that you and Ted, and Doug and his group, independently came up with different forms of word processing.

Van Dam: Yes. I think these were the forerunners of word processing. This is one of the things that Ted was not interested in, because one of the main points of his world view was that hypertext, by definition, was not printable. It was meant to be authored and read on screens. But we had documentation needs, and I figured it was a splendid way to produce documentation, and I could kill two birds with one stone. I had violated the purity of his vision, perverted from his point of view, by introducing all this extra stuff to be able to print things out. We used hypertext in order to link things together, and to assemble documents. We certainly did a lot of that with FRES and our user community faculty members at Brown, and the humanities in particular, got very creative with view specs and macros and printing out their syllabi, and producing books. They were really happy that we had thought a lot about how to make hypertext useful for printing linear documents.

Weber: What did this do to your relationship with Ted, the fact that you didn't stick to his vision?

Van Dam: Well, it certainly was problematic during the design of the hypertext editing system, but since he didn't have an implementation team, in a way, he was stuck with us. We didn't do what he wanted us to do in many ways, but at least something got done. I think for a while, he too took pride in what we had accomplished, but I think he came to realize how little of his vision actually was shown by that first pancake implementation, and has disowned it since. In fact, he incorrectly blames it for the design of HyperCard and other one way [of] linking primitive systems, which eventually led to what he considers the damaged vision of URLs [uniform resource locator]. I've written him and told him that he's giving us way too much credit for having influenced the designers of HyperCard and other systems, and Tim Berners-Lee. First of all, we had two-way links by '68 in FRES, and none of hypertext editing system, FRES, or even systems I produced afterwards, were studied by the people who did notecards, or HyperCard, or any of the follow-on systems. But it's one of the places where we disagree.

Weber: You stayed friends?

Van Dam: For a while we weren't friendly, but then at the hypertext conference in '87, we made up. Ted gave wonderful talks at Brown and at the Bush symposium, which I ran in honor of the 50th anniversary of *As We May Think*. He gave a wonderful talk, and then this is one of those things where you say the gods were vengeful. The morons at MIT; I had hired professionals at great expense, because I knew I was going to have the crème de la crème of the computing and hypertext world there, and I didn't want to have amateur night with video recordings. I hired, at great expense, professionals, and they recorded over Ted's talk. Of all the people to pick! But I have to say, in this particular instance, Ted did not have a hissy fit. He clearly was very discomfited. I was beyond mortified, and he was a real gentleman. He gave the talk again. It was a different talk. Some of it was the same, some of it was different. It was a terrific talk, and it's still available. He really was a pro with that, and we've had some conversations since then. Recently, he's sort of gone back to a very bitter view of the hypertext editing system. I just read in his new book that's coming out what he has written, and it's sort of his old view about what a mistake it was, and how it set the world on the wrong course. He's wrong about that. I will write him again, telling him about

what a nice job he did on this Englebart celebration with his five minute pitch, and how I wish we could get beyond the past and go back to having a normal, respectful relationship, because I admire him and respect him enormously. He and Doug were the two great visionaries after Bush.

Weber: Then at the time, when you formed the company, or you started to form a company around FRES—

Van Dam: Well, I wouldn't say around FRES. Around some kind of hybrid of the ideas that we had implemented and new ideas. It would have been a completely different system.

Weber: What year was this?

Van Dam: '68, '69.

Weber: But that's when you drifted apart with Ted.

Van Dam: Yes, right. Ted was not interested in continuing to work with us for a variety of reasons. The four of us working together had not ended happily. We agreed that there was no further point in working together.

Weber: The four was you, Ted—

Van Dam: Don Ruley, and his boss, Sam Matsa, who at that time was still the director of the New York Scientific Center for IBM.

Weber: But the three of you continued to—

Van Dam: Oh, Matsa and I are still friends. Most recently, he and his wife stayed at my house. We communicate with each other. He's been retired for many years, but yes, that has been a great relationship.

Weber: Then back to your day job; what was going on with graphics?

Van Dam: The whole hypertext thing was definitely a passion, after I had my taste with the first system, and I continued working on it. The two strands started merging, that is, the hypertext strand and the graphic strand, because FRES did have simple line graphics capabilities. Then the next system was the electronic document system, which we wrote up, and transactions of graphics in the early '80s. It was all about graphics, with very little text. It was meant to allow people to construct online maintenance and repair manuals, and other kinds of technical documentation, where you would use the hypertext capabilities to make different trails through the documents, depending upon your need, depending upon keyword settings, your background, whether you had been through those pages before, etc., etc. They were, for the first time, taking advantage of what was then a new technology: color raster graphics. We have beautiful, full color images of sonar documents that we modeled in our system. We worked with folks at Raytheon who are in the sonar business. All of our examples came from the world of sonar electronics.

Weber: Let's pause for an instant. You were linking images to images.

Van Dam: Linking images to images with minimal text and trying to make visualizations of the directed graph that hypertext makes. Trying to have conditional paths based on whether you'd visited a particular site before. There were a number of user interface innovations. We built the modeling system for creating the images, so this was the system that Randy Pausch worked on, where he learned the fine art of demoing and spieling that stood him in such good stead. He implemented a bunch of things in the system, and he was, as I said for his eulogy at his memorial services at CMU earlier this fall, he was the first person in our group who really was passionate about the user experience, and wanted us to really focus not so much on the functionality, but on exactly what the user would be able to do, and how to express it, and how things would be displayed. He was really good at that.

Weber: And Norm Meyrowitz also worked—

Van Dam: Norm worked on that, right, and that was also the time that we started thinking about spinning off the hypertext work, because it was clear to me that there was a full agenda on graphics, and there was a full agenda on hypertext. At that time, I was investing a lot in bitmap graphics workstation technology. Brown was the first university to really commit to the use of workstations for education. I was chairman of the department of that time. I was the founding chairman, starting in '79, and we had very ambitious plans for what was later called the electronic classroom, which is an idea that Bob Sedgwick had had, based on work done by another friend of mine, Ron Baecker, on algorithm animation. Bob, along with his PhD student, and my undergraduate student, Mark Brown, built BALSA, Brown Algorithm Simulator and Animator, that let you see, let's say, sorting out-- sorry. Ron Becker had done "Sorting Out Sorting" as a movie, and Bob's idea was that with workstations running at a half MIP [Million Instructions Per Second], which was huge at the time, and wonderful bitmap graphics, that you could do all this in real-time with user input. He and Mark Brown designed this system, which I used in my introductory course, and Bob used in his data structure and algorithms course, where you could see code being stepped through, and right next to the code, there was a window that showed you a drawing of the data structure and how it was being modified in real time. That became the electronic workstation lab, the electronic classroom. I competed against CMU and MIT to get a major investment from IBM. We were called, when I was finally successful in negotiating that, the three workstation schools, CMU, MIT and Brown. We had this electronic classroom at that time, which was a specially-built ramped auditorium, populated by 40 Apollo DM-300s or maybe DM-100s. I don't remember the model number. No one had seen anything like that before, and we were using that live during these courses for teaching kids concepts. It was beautiful. Norm was there at the time, as was Nicole Yankelovich and multiple other of my students. Once we got the IBM grant, it was clear that I couldn't do that with just my group of mostly undergraduates and a couple of graduate students, and no full-time people. With Bill Shipp, who at that time was running computing at Brown, we came up—he, Norm and I—with the idea of forming IRIS, Institute for Research, Information and Scholarship. It was IRIS that got the big $17 million equipment grant from IBM. Not just equipment, but staff as well to essentially bring 600 workstations to the Brown campus, so that every faculty member would have a workstation. That was the vision. There was a beautiful cover of the *Brown Alumni Monthly* in '84, that shows a mess of workstations coming in through the Van Wickle gates, like new students.

Weber: I think I remember that.

Van Dam: That was the vision that never came to pass, because IBM was unable to make the hardware robust enough and the version of Unix never was robust enough. After a while, they were viewed as boat anchors, and we never took delivery of the full set. In fact, a lot of development at that time happened on the Apple platform. We started with LISAs [Local Integrated Software Architecture], and eventually moved on to other Apple products, and were very influenced by the object oriented work that had gone on at PARC [Xerox’s Palo Alto Research Center]. That's a whole separate story that you will have gotten in great detail from Norm Myerowitz.

Weber: Particularly Intermedia.

Van Dam: Yes. I can't say that outside of having exposed Norm to the three generations of hypermedia systems done under my direct supervision, I won't make any intellectual claim on design ideas in Intermedia. I think that was primarily Norm and his people. I certainly critiqued and had suggestions, but he deserves the credit for the architecture. We can have a separate conversation on all the new ideas that Intermedia introduced into the hypermedia world.

Weber: Stepping back with the education threat though, you mentioned that FRES, you did use in the classroom.

Van Dam: FRES was used by a lot of faculty members, but not in the classroom. What we did do, in the mid '70s, was to use it for courses, but in a lab, because we only had one terminal that had the full power. It was an Imlac, which was like an overgrown PDP-8 basically. It had a large screen, which we divided in windows, a '76 movie, in which we used the term "windows."

Weber: That was a question I forgot. The windowing, when did that—didn't come into any of these early—

Van Dam: Well, Doug lays claim to the fact that he split the screen up into portions, which were forerunners of windows. We had more general windows. They were rectangular. You could have multiple ones open. You could choose various scanned configurations. Norm built an even more general windows system than what we had for FRES, that could be used for arbitrary applications. We were very early in the movement to have windowing in the modern sense of being able to choose what size they were, and having multiple arrangements of windows. We did not have tiled windows, à la PARC, because for a while, we didn't have bitmapped graphics workstations. The Imlac was a vector display, but it was software characters, so you could make Greek fonts and Hebrew fonts and all kinds of other pretty stuff that we couldn't have done with the hardware character generators of the time.

Weber: So you were in the lab with the—

Van Dam: We had a special room where we had this IMLAC terminal, which was hooked to the mainframe. Students would come in to do their work, so the first course in which we did this was "Man, Energy, and Environment." I got a grant from the Exxon Education Foundation to do that. Then the year after, I got a grant from NEH [National Endowment for the Humanities] on probably one of the very few, possibly the only, computer scientist who ever got an NEH grant. I got it from Roger Rosenblatt, who later became a quite well known editor and opinion piece writer for—I think he started off with *Life* and then went to *Time* magazine. That was one of his many jobs. In any case, I convinced him that my

collaboration with this great English semiotics teacher should be funded, and that we should run this experiment of doing a poetry course with all the kids doing—oh man, this is ridiculous. Bob Scholes. You're going to have to edit this out. I cannot be having a brain fart about a collaborator. Bob Scholes, and there was a teacher of writing named Van Nostrand, who also was involved. There's a very nice documentary that we shot with NEH money on 16 millimeter film that I can get you copy of sometime, in which we described the poetry experiment. The beautiful thing there was that these kids in the FRES group, and there was also a control group, that mimicked all the facilities of FRES, but they did it on paper. Then there was a standard group that went to the library and did things in a completely normal way. We compared how they did, and our group had fewer cavities, statistically significant amount. The thing that Bob Scholes loved is that they wrote something three times more than the other kids did, because writing was so easy if you had a computer system to help you do it. As I mentioned in my tribute to Doug yesterday, I believe we had the first scholarly online community. Not the first online community, because arguably, the bootstrap thing had been working with NLS as the network information center for the ARPANET [Advanced Research Projects Agency Network], for years before then. But we were a genuine scholarly community, where the whole idea was to do interpretation of readings, and do recursive commenting of those readings. It was a beautifully run experiment, so I'm very proud of that. In any case, that was still primarily text oriented, with a little bit of line graphics thrown in. Then we got the bitmapped graphics displays with their ability to do color. The graphics became sort of the principal focus in that system, and then by Intermedia time, Norm had this very object oriented view of how documents should be indefinitely nested, so you could have text within graphics within text. I think video was in there as well, and then there was a better balance between text and graphics.

Weber: But the navy project was—

Van Dam: At first, very heavily graphics oriented, because bitmapped raster graphics was just coming into commercial realization at that point.

Weber: So you had links from essentially icon to icon—

Van Dam: Yes, from page to page and from object within a graphical page to an object in another graphical page. Sure, you could have textual explanations, but the text wasn't the focus. There wasn't any outline structure, for example, à la NLS.

Weber: Were you doing other graphics work that was not—?

Van Dam: Oh, yes, absolutely. There was other graphics work going on. We were heavily invested in learning how to do rendering, image synthesis, and building geometric modelers. For example, there was what is called a CSG, constructive solid geometry, system, which was mathematically quite sophisticated.

Weber: When was that?

Van Dam: That was in the mid-'80s. We did volumetric rendering. There was an early version of the marching cubes algorithm that we did, and arguably, that was the first time that algorithm had been practiced, even though GE [General Electric] is associated with that algorithm, because they published first. There was a lot of work on geometric modeling and rendering for 3D graphics that was going on. We built animation systems, so sort of modern 3D graphics was something we were working on at the time.

Weber: Starting in the '70s?

Van Dam: Well, even in the '70s, we were doing graphics, because the configuration that we did our graphics on was a homebrew multiprocessor, which we started in the '60s already, the late '60s. We built matrix multipliers, so that we could do homogeneous coordinate transformations of initially 3D objects. Andy Markowitz, who became quite well known, through a book, I believe, called *Take No Prisoners.* I'm sorry, I'm spacing on this. By Ferguson, I believe. It was a book describing how his company, funded by Matrix Partners, of which Andy Markowitz, who built my first matrix multiplier, was now, by that time, a partner. How they eventually sold Front Page to Microsoft. It talks about the whole story, and Markowitz does not come out as a particularly lovable character, but he's still a good friend of mine, and I think he was a bit maligned in that book. But never mind. I wasn't there, so I've no way of judging. In any case, it's interesting that Andy got his start building that matrix multiplier. He then was one of the key people at Apollo, and we were the first people to buy a lot of Apollos and use them for education. That was an interesting connection. After that, we bought two micro programmable meta-4s, made one a CPU [central processing unit] by providing its instruction set, and the other was a GPU [graphics processing unit]. We provided it with a special purpose, graphics processing unit and instruction set. We called it a DPU, display processing unit, but today everybody called that a GPU. In any case, it had a very sophisticated instruction set, which we would customize on these two machines, shared with 64KB of memory, so it was a symmetric multiprocessor in some ways.

Weber: What year?

Van Dam: This was in the early '70s, and we had that attached to the mainframe, and we did experiments in migrating code in real-time, from the mainframe to this multiprocessor, to do load balancing. Then we used a commercial 3D graphics engine, called a Vector General, which had its own transformation hardware, but we had a microprogrammed unit, which was much more sophisticated and much faster, and could do not only 3D transformations in real-time, but also 4D, because we worked with a colleague of mine named Tom Banchoff in the math department, who was very interested in four dimensional geometry, hypercubes and other hyper surfaces in four-space.

Weber: How would you render that?

Van Dam: By doing 3D projections of 4D objects. He became an expert in slicing and dicing through the four dimensional volumes to show what their three dimensional projections look like. So in the way of thinking about this, if you take a cube, you can slice it either like this, and you get squares, right? Or you can slice it on the diagonal and get triangles, and then hexagrams. No, you certainly don't get a square cross section. I have to think about that. I'm not that great at visualizing which is why I've always been so profoundly attracted to graphics. In any case, he would slice—no. I'd have to draw it in order to see it. He would do this for four dimensional figures, and he would make movies with a PhD student of mine who graduated in the mid-'60s, Charles Strauss. Banchoff and Strauss Productions and their movies were widely known in the mathematics community, because they show the slicing and dicing of four dimensional figures. That was all shot off our equipment. That was a very sophisticated multiprocessor, and we did distributed computing. In fact, I organized the world's first workshops on distributed computing, and ran them at Brown, and published the proceedings, because I had my students do real-time transcription off what everybody was talking about. I was very interested in all the systems aspects of graphics from the hardware on up, and including distributive processing. We had a remote procedure call mechanism that allowed us to have these heterogeneous machines talk to each other over the 50 KB link.

That was a very sophisticated system, and just like NLS, shoehorned into 64KB of memory, with a 1 megabyte bladder. It was amazing what you could do in those days. We have some gigabytes of memory, and things don't move any faster than they did when we had megabytes or even kilobytes of memory. It's amazing. So yes, we were doing a lot of graphics from the ground up, most of it 3D graphics.

Weber: We're getting to the end of this tape. SIGGRAPH, is there any other thread to follow there?

Weber: No, I don't think so. We essentially created a forerunner of SIGGRAPH, but the credit should go to the people who put on in the first conference at Bowling Green, and that kept expanding the vision, and kept getting more and more people to come and more equipment manufacturers to come and put on this amazing set of shows, year after year after year. SIGGRAPH is a huge volunteer organization, and it's quite amazing how well it works.

Weber: You had a story about Steve Jobs' reaction to networking.

Van Dam: Oh, right, yes. Andy Herzfeld was one of my students. He did a very nice final project in the introductory graphics course that used that [our] multiprocessor system, which was called BUGS, Brown University Graphics System. It was a spline editor, where in real-time, you could adjust the Bézier spline control points and shape the curve. I had not had much contact with Andy, and Andy, of course, was very active in the homebrew club and the whole west coast personal computing phenomenon. Then one day I got a call, and I don't remember whether it was from him or from Steve directly, that they wanted to come to Brown. They knew about our very early on, aggressive use of workstations for teaching and research. All the faculty had workstations and our freshmen had access to workstations, which is something that not even MIT and CMU had for quite a while. They knew about that, and they wanted to tell us about a development that we might find interesting. That turned out to be the Mac. It was the early Mac, the 128KB Mac, where you had to milk with the floppies. Didn't have a hard drive. No networking, no client server architecture. The provost had been invited to come see the demo and whatever it was, and to have lunch with Steve Jobs, who already was a pretty well known guy, and we knew and really liked the LISA a lot. We were expecting something that would be the follow on to the LISA and would be bigger, better, faster, cheaper. In some ways, the Mac is, and some ways it isn't. But the fight that I got in with Steve, in the provost's presence, and both he and I have tempers and they flared, was about Steve's vision of personal computing. We had been doing personal computing, a person sitting down at a workstation, but not giving up anything, because à la PARC, the Apollo folks had implemented client server architecture, and they had a very good networking protocol and this object oriented operating system called Domain, which was arguably well ahead of Unix, but eventually lost out to Unix. That's a separate thread. I can tell you more about that if you like. In any case, this was a really sophisticated system, and we were doing real-time graphics and text and the whole nine yards. I loved the Mac and its clean crisp display, and its real-time stuff, all done in clever software that Andy Herzfeld and Bill Atkinson, who's a real wizard—they both are, but Bill is amazing. Bitblit and other more clever extensions of Bitblit. I thought it was terrific, but I said, "How do we connect these in a network?" Steve said, "You don't." I don’t know whether we had the fight about the hard drive first, or whether the networking fight came, but those were major points of disagreement, and Steve said, basically, "No need for that. You're supposed to work by yourself. It's your personal computer. Small enough and light enough that you can pack it in this case." They had one of those very early soft packs for the Mac and that was his vision. You have this nice, cute, little machine and you do your own personal work on it. It was not part of his world view that people should work together, and that they should share expensive resources. He learned differently rather soon thereafter, fortunately for all of us.

Weber: Well, thank you very much.

Van Dam: My pleasure.

Weber: It's an honor and a pleasure. Thank you.

Van Dam: Thank you.

END OF TAPE 2

START OF TAPE 3

Van Dam: Over here's another thread which is graphics standards. In '71, I had my first sabbatical and took a whole bunch of my graphics group, all students to my home country, the Netherlands, and we spent a very wonderful year in Nijmegen, which is in the southeast of the country, and we set up an instant graphics group there. I taught both introductory computer science and computer graphics, and we got a very high end state-of-the-art PDP-11/45 and a Vector General display, and for a while, that was sort of the high-end standard for research groups and we proceeded to do a lot of programming. I was familiar at the time with a system called GINO Graphical Input/Output that had been done in the UK at the Cambridge Computer Aided Design Center, and it was kind of the FORTRAN [IBM’s Mathematical Formula Translating System] idea of abstracting physical devices to become logical devices. I thought we could make GINO into a much more interactive kind of system that would deal with all kinds of peripherals, which could include mice, and joysticks and cursor control keys, and it could run just like GINO did on a variety of platforms by abstracting out both the display surface and devices to become logical devices à la Fortran. With my students and forming alliances with Delft Technical University and with Cambridge Computer Aided Design Center, Peter Wood’s group in particular, and Peter Wieringa in Delft, we set out to design a system that came to be called the general purpose graphics system, GPGS. It was a device-independent system with logical devices that includes both the interaction devices which we classified into five categories, and the display surface, itself. GPGS had a scene graph, so it had an internal hierarchical data structure which was certainly a descendant of the sketchpad hierarchical data structure. We published it and we thought this really works pretty well on a variety of devices including even storage tubes, which are not exactly real-time graphics displays. GPGS eventually became the standard of Scandinavia, that was very nice, for computer aided design shipbuilding and things like that. In '74, I helped organize a program on the idea of standardizing graphics library at the National Bureau of Standards that was run by a former student of mine named Ira Cotton, who had been my undergraduate student. And a bunch of people working in computer graphics converged there for a 2-day meeting. It was a time to think about standards and graphics; how would we go about it. NBS, National Bureau of Standards, was the natural host for that kind of thinking. In an evening birds-of-a-feather session, I remember this vividly, some number of us, including former students of mine who had worked GPGS like Dan Bergeron, who later became Department Chairman of UNH [University of New Hampshire], and still, to this day, works in computer graphics. We said, you know, we've learned enough about GPGS that we think we can base a standard on it. That was again debated at an international workshop run by Richard Guedj in France where Alan Kay and Nicholas Negroponte and Alan Newel—not Alan Newel, Martin Newel—and some of his colleagues from the Cambridge group came. There were some of us who very much were in favor of trying to standardize, and some people who were cynical or skeptical. Negroponte and I had a very public disagreement about whether it was time to standardize, he taking the negative and me taking the positive.

Weber: What year?

Van Dam: '76.

Weber: At the conference.

Van Dam: At that workshop in Seillac, France. To cut to the chase, Bert Herzog, an old friend with whom I had taught graphics in the summer courses, who unfortunately passed away this past summer, and Bob Dunn, who had been a sponsor of mine at the same time that the work was going on for ONR [Office of Naval Research], so he was the army and was co-sponsoring some of the work on electronic documentation using hypermedia. Bob Dunn and Bert Herzog convinced the graphics establishment that there should be a standards effort endorsed by SIGGRAPH that became the Graphics Standard Planning Committee. We, with a small group of people that met mostly at Brown University designed something called the Core Graphics System, which was based very largely on GPGS. The Germans took an early draft and under another dear friend, Jose Encarnacao's leadership built a derivative that was 2D at first called GKS, which in German, Graphical Kernel System—means core, kernel, core. It was misunderstood that these were competitive systems and they certainly diverged after awhile, but they were based on the earlier core draft, which in turn was based on GPGS, which we did as a three-university project with most of the work happening in my group in Nijmegen. We tried to get Core to become the national and then the international standard. It had pretty good reception at first, but eventually, it became an ISO [International Organization for Standardization] standard. Then it actually morphed into something called PHIGS, Programmer's Hierarchical Interactive Graphics System. PHIGS went through the NC process and the ISO process, and I learned that what we had been able to do at Brown in a year, essentially, as a tiger team meeting often and rapidly iterating our understanding of how to do this, and not solving all of the world's problems by any means, but getting something done, we were able to present the design in '77 at the SIGGRAPH conference in San Jose, where I got thrown in the swimming pool *<laugh>* by my friends who had worked on that committee. In any case—

Weber: Was that a positive or a negative?

Van Dam: Well, it was a very mixed blessing because I had my wallet in my pants still, since I was completely unprepared for this little *<laugh>* bit of good fellowship, and I had ruined pictures of my kids and my driver's license and some other things. But never mind. It was well meant. It was a good practical joke. In any case, I've almost forgiven them, as you can tell. *<laugh>*

Weber: It's been what, 30 years?

Van Dam: This was before Foley and Van Dam, the book. I then saw that informal standards can be done quickly by a small committee, especially a committee in which there really weren't any commercial interests to speak of, that once you got involved in a formal standards process, it became heavy weight and you had people who represented what their companies would find useful, and that was a lot harder work and I didn't have the stomach for it. Jim Michener, one of my other early PhD students, bless his soul, participated in that official standards effort, as did multiple other people who had worked on the original Core Project and man, they labored hard to try to get an international standard. At about that time, SGI [Silicon Graphics, Inc.] was starting to really become a force, and they had a completely different idea about what a graphics library should look like. They controlled that graphics library and they became a dominant producer of high-end graphic systems, and so GL [Graphics Library] and then OGL

[Open Graphics Library], for open GL, eventually took over and it showed very clearly that a package designed by a small group and continuously refined by testing with a large customer base can win out over the political convergence process; that is the standards process. Obviously, standards become camels, i.e. horses designed by a committee. There are a lot of compromises that you have to make in order to get people to agree. People sometimes argue too narrowly for their company's or institution's interests, and so effectively, the graphics standard never really became a commercial force to be reckoned with. OGL did and RenderMan did, all packages designed by much smaller groups of people without the constraints of the political consensus process of official standards. That was kind of a negative thing, but on the other hand, we learned a lot from doing it, and some of the ideas have survived. For example, Ingrid Carlbom's viewing pipeline survives to this day. I was able, when I took a sabbatical working for Bill Poduska's third company, which was a graphics supercomputer company initially called Stellar, I reconvened a bunch of my old friends and some new people, including folks from competing organizations, and we designed an extension to PHIGS to bring it into the era of modern rendering, which was called PHIGS PLUS. For a while, it looked like PHIGS PLUS would get traction that PHIGS hadn't, but it never was able to stand up to OGL and that's fine. Certainly differences in design philosophy, and interestingly enough, PHIGS and PHIGS PLUS had built-in scene graphs, because they were variations on Core, much more sophisticated than Core. Core was a more sophisticated version of GPGS, which had had scene graphs. OGL, although initially in GL, there was a scene graph, that has faded completely, but game engines all have scene graphs. This old religious dispute about should you or shouldn't you have a scene graph just keeps going around and around. There's a wonderful paper by Myers and Sutherland called *The Wheel of Reincarnation* about how the same old ideas come around and around and just have different implementations. That certainly is the case for graphics libraries and graphics standards, and the scene graphs are healthy and alive. Some people like them and they're useful, and some people don't like them and want immediate mode. So that will be a continuing battle and people should have choices.

Weber: But it's some of the way things come around due to—well, you talked about some the other night around the Engelbart anniversary that there's very little education about the past in computer science.

Van Dam: Yes, you make a very good point. Certainly, things are rediscovered, and people don't read any of the old papers, and don't know that some of the ideas are 30, 40 years old and that there were trade-off analysis about how things might be done, and the pros and cons of the different approaches. They reinvent and sometimes what they reinvent is arguably better than what was there before, and sometimes it isn't. This is not a culture in our computer science community that prizes a lot of reading of the past, and that is definitely one of the things that is wrong with what we do. We're so hyperactive and so focused on making improvements and trying to do it on the Moore's Law timescale or faster, preferably, certainly in graphics way faster than Moore's Law. New results every six months in the GPU industry and at SIGGRAPH, there's huge progress from year to year. I think there's bound to be some rediscovery, but I think in all fairness to the people who are making the advances, most of it is in the thinking and taking advantage of new capabilities and the field is thriving and having amazing success.

Weber: A couple of miscellaneous questions. When you saw the Engelbart demo, you were working on similar things.

Van Dam: Yes, very much so.

Weber: Did you feel disappointed in any way that something was so developed in what you thought was a new field?

Van Dam: Yes, of course. I mean, I was the country mouse <*laugh*> and I went to the big city. As I mentioned yesterday, here I was working with a team of three undergraduates working part time, and there Doug was with his far more developed system philosophy, and system build from the ground up with 17 people, and this amazing projector, and a link to a remote site and it was mind-blowing. I'm not ashamed to say I was very skeptical of how much of it was real and how much of it was demoware, whether this was really a productions system, or whether they had been very lucky in cobbling something together. It wasn't until I spent time onsite that I realized that yes, this was a very deep vision and not just some showmanship. The showmanship was magnificent, by the way, and I have to have profound respect for it, but it's, I guess, equivalent in the sports world, you're really proud of yourself for having run an under 6-minute mile, and then you go see somebody who’s doing an under 4-minute mile and you say what was I thinking? <*laugh*> What was I doing? What right did I have to be proud of what I'd accomplished? So imitation is the sincerest form of flattery, and I imitated what I thought were the best features of Engelbart's functionality in NLS and in FRES. We copied a lot of the ideas, even though we implemented them in a different way, and also kept some of our old ideas from the first system.

Weber: Did you ever have moments like that in graphics?

Van Dam: Oh, you always have moments in any field where you see yourself lapped by somebody you didn't even know about, somebody who came out of left field, was doing something that they hadn't published before, and you look at it and you say wow, that's really cool. I wish I'd thought of it. But I think we did enough things that were novel over the last three or four decades that I feel okay with occasionally being lapped. That's all part of the game.

Weber: When did you add networking to your own hypertext systems?

Van Dam: We never had networking as an explicit component.

Weber: But you had timesharing sort of things.

Van Dam: We had timesharing in the sense that, just like with Engelbart, we had a monitor that allowed multiple, relatively underpowered glass Teletypes, and somewhat fancier stations to be hooked to a single mainframe, but that was not networking. We certainly made use of virtual machine technology that IBM had available as early as '67, because we bought [a] 360/67 at Brown. In fact, one of the things I'm really proud of that was a tiny fraction of the mother of all demos is that in '68 or '69 in Boston, we had the hypertext editing system running on a 2250 model 4 over a Bisync link to a mainframe in Cambridge using an early version of their virtual machine operating system which was called CP/CMS [Control Program/Cambridge Monitor System]. We were very early adopters of virtual machine technology, courtesy of Cambridge. That center too disappeared, but the idea certainly didn't. Everybody now uses virtual machine simulators. But IBM invented all of that in Cambridge in the '60's. We had minimal networking in the sense that we used this high speed Bisync link and the IMLAC ran over [a] 1,200 baud modem, but it's not networking in the modern Internet sense of networking. That didn't exist. That had to wait for the ARPANET and the NSFNET [National Science Foundation Network] and the development of modern networking. By that time, I'd stopped working effectively on research systems, and in '90 was one of the founders of Electronic Book Technologies, which was the first company that unified the idea of

declarative markup through SGML [Standard Generalized Markup Language]. That's a whole separate and long story. SGML is what XML [Extensible Markup Language] is getting back to, infinitely more sophisticated than a particular tag set, which is HTML [HyperText Markup Language]. In any case, it unified declarative markup for documents with hypertext ideas, and so it was based on all of the hypertext systems, including Intermedia that we had developed at Brown. It was sold commercially and the company lasted for about ten years. That was a really neat set of people and technical achievements.

Weber: When you look around today, how different is it from what you imagined?

Van Dam: Oh, that's a good leading question. Back in the NLS mother of all demos days, none of us, except possibly for Doug anticipated this sort of worldwide access that we have today.

Weber: Ted?

Van Dam: I don't think Ted anticipated worldwide access. I think later, he probably did, but in the beginning, we didn't talk about that because there was no existence proof that that could be done. The whole idea of packet switching only came on in the late '60's. We were contemporaneous with early networking ideas and packet switching, but we certainly didn't have an ARPANET connection and get used to that kind of thinking. I think the idea of having everybody in the world have instantaneous access to information, that would have been pretty science fiction-y, and the idea of a real Dynabook of the type we all carry as laptops or, better yet, tablet PC's, that was pretty far away. I heard Alan Kay talk about Flex and Dynabook, I thought, “Yeah, yeah, yeah, it would be lovely,” but we were still in the mainframe area. Minicomputers were just starting to be common and that was a huge improvement. But to go from there to microcomputers and then from there to laptops, well, that was a stretch, although I have a lovely plastic mockup in my office of a maintenance and repair technician's toolkit of the future which is, in effect, this big. It's a color screen and it was meant to be touchscreen. It would run that electronic document system that we talked about earlier. And the idea would be that you have all of that power in that portable form factor, and that was mocked up in 1984.

Weber: A tablet sort of?

Van Dam: Yes, sort of like a large color tablet.

Weber: Networked or wireless?

Van Dam: Oh, we didn't worry about the networking. We figured something like that would come along. But it had a slot for it all mocked up. This was done by an industrial design student at Rhode Island School of Design. We thought about little what you would now call a flash drive or a thumb drive. The idea was that it would be screen-centric in that you, instead of typing, you would point and click. That was fully thought out, so at some point, we woke up to what was happening with personal computing and and saw that this was all going to be possible at some point. In the '60's, no way. Not me, for sure. Doug, who knows?

Weber: Thank you so much.

Van Dam: My pleasure.

Weber: It's been a pleasure and an honor.

END OF INTERVIEW

