An Interview with

GEORGE KAROLY and MARCEL DAYAN

OH 446

Conducted by Thomas J. Misa

on

21 November 2013

Control Data Corporation History Project

Melbourne, Australia

Charles Babbage Institute
Center for the History of Information Technology
University of Minnesota, Minneapolis

George Karoly and Marcel Dayan Interview

21 November 2013

Oral History 446

Abstract

In November 2013, CBI director Tom Misa conducted a series of oral history interviews with 13 former employees of Control Data Australia (1963-89) including the details of each person's career, before and after working for Control Data. Topics that are common to many of the interviews include Trevor Robinson's key role in organizing Control Data Australia; the early computer sales in Australia to the Bureau of Census and Statistics, Department of Defence, Postmaster General, Commonwealth Scientific and Industrial Research Organisation (CSIRO), Bureau of Meteorology, and several prominent Australian universities. Control Data Australia did business data processing for such large concerns as Broken Hill Proprietary (BHP), Telstra, and others. A distinctive emphasis was its work in developing computer systems for race-track betting for the state of Victoria's Totalisator Agency Board (TAB) as well as for other Australian states and New Zealand. Other topics include relations with Control Data's headquarters in Minneapolis, business data processing, data centers, database management, networking and Cybernet, and projects done in several Far East countries.

Interviews were conducted with Richard Bament, John Baxter, Ron G. Bird, Tony Blackmore, Lyle Bowden, Marcel Dayan, Ian Downie, Julie James, George Karoly, John O'Neil, Garry Pearce, Rob Robertson, and Bruce Wilson.

Misa: My name is Tom Misa. It’s 21 November 2013 and I’m here in Melbourne, Australia, talking with George Karoly and Marcel Dayan. I think we’ll start with George because your experience with Control Data Australia, George, goes back quite a way. You’ve given me your CV so we have many details there, but maybe you could give a brief overview of how you entered the computing field.

Karoly: I was in the electrical engineering school of the University of New South Wales, it was then the New South Wales University of Technology; its current name is University of New South Wales. I was a lecturer in electrical engineering there, and shortly after I joined them, they installed an English Electric DEUCE [Digital Electronic Universal Computing Engine] computer.

Misa: English Electric DEUCE?

Karoly: Yes. Do you want any details on that? Or you can look it up on the website.

Misa: I can look it up, but I’d be interested in your personal experiences with it; what you might have used it for, your vantage point.

Karoly: At this stage, I was interested in computers but I knew virtually nothing about them except what I’d read in a few articles. The DEUCE had a main memory of [mercury] delay lines, and backup storage of magnetic drum. The delay line storage was about 400 words, the drum storage about 8,000 words. It was a binary serial machine with 32-bit words and there was virtually no software. You had to program it in its machine language, which was quite a complex, two plus one address structure. The way we used to prepare programs was: we would fill in sheets with instructions, which would be punched on cards but not in a decimal form, they were punched in serial binary.

Misa: Serial binary.

Karoly: Every row on the card was an instruction in binary form. This process, as you can imagine, was fairly error prone. There were some, what I would call subroutines supplied with the machine, including a set of what were called matrix interpretive programs. Strictly speaking, they weren't matrix arithmetic. What they did was perform arithmetic operations that you desired on arrays of numbers. In other words, say you have two rectangular arrays of similar dimensions, A and B, and you wanted to multiply every element of A by the corresponding element of B, you just told it to multiply A by B.

Misa: It would do that matrix multiplication, then.

Karoly: But it was not true matrix multiplication. A matrix multiplication multiplies rows of A with the corresponding columns of B. If you've got true matrices, the multiplication goes this way. This program took corresponding elements of the two and performed whatever operation you wanted. For instance, you could say I want the square root of matrix A, and it just did the square root of every element.

Misa: Each element.

Karoly: Yes. This was quite useful but very slow. I mean, as you can imagine, overall, the machine was very slow although its internal speed was quite high. Its clock ran at a megacycle. So that the cycle time was 32 microseconds and it could perform addition in approximately two cycle times. But a multiplication took — I can't remember — considerably longer; might've been as long as a millisecond. Division was even worse because there was no hardware division so it was programmed division.

Misa: Program the thing to do division, the tricky part there.

Karoly: And, you know, I enjoyed programming for it. It mightn't have been very efficient, but it was a very interesting intellectual exercise.

Misa: Fun to do.

Karoly: Probably the biggest program I wrote for it was an analysis of a multi-element microwave array used for instrument landing at the Sydney Airport. The mathematics was simple enough. Being an electrical engineer, I knew the required formulae, but making the computer do it was a slightly more difficult problem. In fact, I discovered for myself the process for using flow charts. The first time I tried writing the program, after a few days I got completely lost. So being an engineer, I thought hey, surely the way you do it is you do an overall plan and then break it down into details.

Misa: So the standard boxes, or however people schematize it. Yes. George, I was just looking at your CV. Very early on, you were at the school of electrical engineering in 1954. This is an early time in computing.

Karoly: Yes, I gave a few lectures on ALGOL and COBOL. The DEUCE did not have either of those, but I gave some lectures at Sydney University.

Misa: That was at Sydney University?

Karoly: At Sydney. To put it into historical perspective, Sydney University also had a computer. It built its own version of the University of Illinois ILLIAC, and christened it SILLIAC.

Misa: SILLIAC, right.

Karoly: But in a way, our machine got rechristened also; it was named UTECOM, University of Technology Computer. SILLIAC, I'm pretty sure, became operational just before UTECOM. The way I got into commercial computing was in early 1960, a friend of mine, who had a management position in Amalgamated Wireless Australia [AWA], one of the top electronics companies in Australia, told me that they just got an agency for Bendix Computer. And he asked me would I like to join them, go over to Bendix, learn all about the computer, and teach their staff. He rang me about mid-January, and told me he'd like me to join AWA in time to be in the States by the end of February. I told him I couldn't leave the university on such short notice.

Misa: You had responsibilities.

Karoly: And he said just leave it with me. A few days later, I got a call from the Vice Chancellor of the university, the top administrator of the university, saying oh, Mr. Karoly, this is just the sort of industry cooperation we want. If you want to, we will lend you to AWA on unpaid leave until you're prepared to come back to the university.

Misa: That's Amalgamated Wireless.

Karoly: So I was in Los Angeles, at Bendix, on the 28th of February.

Misa: 28th of February, wow.

Karoly: [Laughs.] Learning all about their G-15.

Misa: Bendix G-15 was the machine you were learning on.

Karoly: Yes. I realized it'd be a long time before the University of New South Wales got any more up-to-date computing equipment, so I stayed with AWA when they set up a service bureau using this G-15 equipment. They were also selling them, but they had a service bureau in Sydney.

Misa: The service bureau was in Sydney.

Karoly: Yes. I can't remember who all the customers were, but there was a number in Sydney, also a large customer in Melbourne. Maybe they even used it in their own office in Melbourne, I'm not sure.

Misa: So when you were doing the service bureau work, were you doing programming?

Karoly: Yes. Programming commercial systems, and I was in charge of the programming side of the service bureau work.

Misa: You worked there through 1962?

Karoly: Yes.

Misa: And then in May 1962, you were appointed programmer and analyst at the Australian General Electric Company.

Karoly: What actually happened, it became obvious to me that AWA management weren't particularly interested in their computing department, or couldn't understand it, or both. And GE was setting up a service bureau using GE 225 equipment. I have a feeling they had other service bureaus in Australia, but this was in Sydney.

Misa: Sydney.

Karoly: Yes. And I joined them and this was my first experience with a machine that actually had random access storage. The GE 225 had core storage. Not very large; I have a feeling it was 8,000 20-bit words but I won't guarantee it. I do remember wondering what in the heck was I going to do with all that storage. [Laughter.]

Misa: That huge amount of storage; well, at the time, yes.

Karoly: Yes. I quite enjoyed working with GE. I eventually left them because whenever something interesting cropped up, they brought in American experts, experts in inverted commas ["experts"] over to deal with it. And some of these experts knew nothing about what the problem was.

Misa: So they brought in American so-called experts, but where did that leave you?

Karoly: Rather unhappy; so that when I saw Control Data's advertisement, that sounded interesting. Actually, Trevor Robinson interviewed me for the job.

Misa: Robinson was trying to start a whole new company, really, wasn't he?

Karoly: Yes. I don’t know how many people we had, all I know is that my employee number was 13.

Misa: Number 13. So you were one of the first.

Karoly: Yes. I think John O’Neil was before me.

Dayan: So was Ron Bird. Ron Bird was number one.

Karoly: Yes.

Dayan: I believe there was seven of them that started, and then they decided to give themselves numbers, and Ron Bird was number one because his [last name] started with a “B.”

Misa: Alphabetical order.

Karoly: Yes, the first seven. Was it seven? I thought it was eight, ten, but I’m not sure.

Dayan: And George Crawford was number two.

Misa: So “B,” “C,” okay, on down the line. “D” you could’ve been number [interrupted]

Dayan: I wasn't quite that early, I guess.

Misa: So that would be in November of 1963.

Karoly: Yes. I worked for Control Data in Canberra until the end of the year, then I went over to Palo Alto. I went to Palo Alto to the Control Data Software Development Center in January 1964, and stayed there for approximately six months.

Misa: Six months?

Karoly: No, only five; January to May.

Misa: Five months.

Karoly: Yes.

Misa: George, do you have any memories of Trevor Robinson?

Karoly: Oh yes.

Misa: He played such an important role, but how did he strike you?

Karoly: Tend to run out of adjectives. He was very good with people, no question about that. He must have been a good salesman or he would never have sold all those systems. I would guess he was a very good organizer because he started that company up from nothing and made it a very successful company. And personally he was a very good person to work for.

Dayan: Well the fact that we still have those lunches 40 years and after, I think, was really due to his influence and the *esprit du corps* that he was able to engender in the group.

Misa: He had an unusually wide set of contacts and a very strong set of people skills.

Dayan: And I'll tell you an interesting story about it, when we get to my stage.

Misa: Okay.

Karoly: So I think the CV says that I wrote some of the software for the SCOPE operating system for the 3600.

Misa: It says yes, taking part in the development of SCOPE for 3600, and that was one of the earliest operating system projects at Control Data.

Karoly: Yes.

Misa: So you were in Palo Alto where that was being developed, so it was funny, in a way, you were going there for training but then also participating in the writing of SCOPE.

Karoly: Well, yes. Really all the training I had was the machine details on the 3600, and all the SCOPE system. What that particular task was to write the overlay loading process for the system. SCOPE had provision for program overlays stored on tape, and this program I designed and wrote was designed to load and then to store them, as required, at an optimum speed. I haven't got any technical details of that because I don't think I've kept any, unfortunately. The program was named [by me] LOVER, Load Overlays.

Misa: Load Overlays, okay.

Karoly: It might still exist in some form, I don't know. Anyhow, that's what I did and it did work before I left.

Misa: So your contributions went out everywhere that SCOPE went, in some sense. And SCOPE, of course was extensively used.

Karoly: Well, there were many SCOPEs; as many SCOPEs as there were Control Data machines, just about.

Misa: Oh really? Can you say a little bit about that?

Karoly: Well, initially, anyhow. The lower 3000s ended up with a disk-based operating systems, the first of which was called MSOS. But the original tape-based operating system for the 3200 was called SCOPE.

Misa: So you were resident then in Palo Alto for those five months.

Karoly: Yes.

Misa: Was the idea that you'd stay there, or was it always the idea that you'd be coming back to Australia?

Karoly: Coming back to support the installed systems in Canberra. You see, Canberra would've been the headquarters of the major initial customers; the CSIRO and the Bureau of Census. They both would've had [Control Data] 3600 installations, to which later they added that lower 3000s. But initially, they were 3600s.

Misa: Can you say a little bit about the type of work that you did for supporting those two? Did you do programming?

Karoly: A little, but my basic task was to keep the customer systems working and the customers satisfied.

Misa: And what responsibilities did that entail?

Karoly: Well, the staffing was that we had some so-called site analysts resident at each site, and I don't know exactly when it started, but shortly after I got there, I was virtually in charge of the support. What would happen is that any problem that arose, the site analyst would handle it if they could. If he or she could not handle it, it would then be referred to me. There would also be weekly meetings with the customer's top computer man, whoever he was — I don't know what titles they would've had — at the site meeting with me, and I think the site analyst would be there, and the responsible Control Data salesman, to discuss any issues that arose or any problems that they were foreseeing, or whatever.

Misa: Those facilities for CSIRO, as well as for the Census Bureau, were really crucial to launching Control Data Australia.

Karoly: Oh sure.

Misa: So they were really watching those accounts quite carefully. Do you recall any of the successes or any of the problems that occurred?

Karoly: Well, let me just try to think. Bureau of Census was interested in large scale data processing of statistical data. And there was really no suitable software program, and they were developing their own. I can't remember how many years; I can't remember what they called it; somebody else from Control Data might remember. Do you remember the Census software?

Dayan: No. I was never involved with Census.

Karoly: I was involved with it in an advisory capacity, but not in any formal way. They were also interested in, well, what software that Control Data was developing, because at that stage Control Data didn't have a suitable COBOL compiler. They wrote everything in FORTRAN. Later on, I think, they got the COBOL, but initially they wrote everything in FORTRAN. I think this system they were developing they actually wrote in assembler. The assembler, by the way, you probably know was called COMPASS.

Misa: COMPASS, yes.

Karoly: So much about Census. Now, CSIRO, being by its nature a research organization, was interested in developing new software. And they decided that they would purchase magnetic drums from Control Data, and design their own modification. In fact, you could say, design an all new operating system based on drums rather than magnetic tapes. And that system was also going to use display consoles for various other things. And for that purpose, that's the paper from the 1966 Australian Computer Conference.

Misa: I'll just read the title. It's by you and by R.H. Kerr, "The Utilization of Keyboard Display Consoles in a Conventional Operating Environment." It's about the CSIRO.

Karoly: Yes. And I want to show you one of the references, which I highlighted — I couldn't find my copy — that describes the operating system that CSIRO designed.

Misa: Okay, that's Austin and Holden, "The Development of a Drum and Display Monitor," *Proceedings of the Third Australian Computer Conference* in 1966.

Karoly: I tried Google on it, [but] there's no digital copy. There are some library copies in Australia. If you asked them, they'd send you a FAX copy or a digital copy.

Misa: Through interlibrary loan, lots of times it's possible to turn up the original.

Karoly: They've got copies of the original paper. Look, if you Google it, you find out which library has it, probably the National Library in Canberra, but I'm not sure. Anyhow, you can keep this copy, so you've got it.

Misa: Splendid, thank you.

Karoly: I just copied my copy of it. Easy. Put it in the copier and press a button.

Misa: Thank you very much.

Karoly: I won't talk about that because I think that's enough detail about that.

Misa: The drum and display project, yes.

Karoly: Well, about my part of it. I was involved in it more as an observer than anything else in the overall design. But Robin Kerr and I designed that particular subsystem. From memory, Robin went to IBM from us. I think he went to IBM Research.

Misa: Oh yes?

Karoly: Not absolutely sure, but I'm pretty sure.

Misa: Might be able to look that up too.

Karoly: Okay, right.

Misa: In February of 1965 you became Assistant Applications Manager; and in August 1965 appointed Applications Manager for Control Data Australia.

Karoly: The title's a bit misleading. What the Applications Manager — I think this explains it a bit — what the Applications Manager did was twofold: provide software support for customers; and what software support was required for the sales staff. I think

I had a group of about 20 analysts. Anyhow, that paper gives some detail. Well, in fact, I became, if you know Control Data history, you probably know that part, Regional Applications Manager.

Misa: Regional Applications Manager. Tell me how that worked out here.

Karoly: I ended up being Regional Applications Manager for not just Australia, but also for what Control Data called the Southwest Pacific, which is a bit of a misnomer because it also included Korea. [Laughs.]

Misa: Doesn't sound like southwest. So the countries would be Australia [pause]

Karoly: New Zealand, South Korea…

Dayan: Japan?

Karoly: Japan, I assume. I'm not sure they had anything much in Japan. Corporate history will tell you what the countries were. Anyhow, when I was in that position, Trevor said look, you can't stay in Canberra, I need you in Melbourne. So I moved to Melbourne.

Misa: So you moved to Melbourne, okay.

Karoly: And around that time, Control Data started to get involved in doing TAB business, the Victoria TAB, and I can give you a little paper on that, too.

Misa: Just let me read the title; George Karoly, “Real Time Systems at the Victoria TAB and Historical Overview,” and this comes from the *Australian Computer Journal*, Volume Nine, Number One, March of 1977. So the Victoria TAB had a huge impact on many different things; then there’s a connection to New Zealand, as I understand.

Karoly: Yes. I was only marginally involved in the original project, in the sense that one or two of my analysts worked on the original CARBINE software. I certainly got fairly interested myself in it, but I had other responsibilities, as you can imagine. I’m frankly trying to remember how I changed from Regional Applications Manager to — I can’t remember what my title was — some sort of consultant, internal systems consultant, for the internal technical consulting.

Misa: I don’t know whether this is the right chronology, but it says June 1968, appointed as Senior Staff Specialist.

Karoly: Yes, that’s what it was.

Misa: And that sounds to me that if you’re a Senior Staff Specialist, that’s not with a specific region in mind, but rather for the entirety of CDA, is that correct?

Karoly: Well, there was some talk of me going to Minneapolis on a semi-permanent basis, but I was only prepared to go if there was essentially a guaranteed path to return to Australia, some sort of predetermined time scale.

Misa: You didn't want to be transferred permanently to Minneapolis?

Karoly: Well I wanted to bring up my family in Australia. And that just never happened.

Misa: What kind of contact during these years did you have with people in Minneapolis?

Karoly: I was there at least once a year for various meetings. I think there used to be an annual meeting of all the Regional Applications Managers. I used to attend that, and sometimes if there was something specific. For instance, I know I was there in connection with the Queensland TAB. I was trying to find suitable hardware for the system.

Misa: So you'd be there for an annual meeting. How long would those trips normally be?

Karoly: Never less than a couple of weeks. Typically that sort of duration.

Misa: You were traveling there and visiting there, but never working there on assignment.

Karoly: No. As I say, the sort of visit I had was typically in connection with the Queensland TAB, about which Marcel can tell you more. They weren't happy about using the 1700 . . .

Dayan: They were System 17, by that time.

Karoly: There was the System 17 for internal controlling hardware. I went to Minneapolis to see if I could find something better. And I could, but they wouldn't let me have it.

Misa: What was it?

Karoly: I can't remember; can you remember?

Dayan: I reckon it's the one that I ended up getting for New Zealand TAB, the MP32. It wasn't in the Red Book; it was in the extra pages; it was in the Special Product section.

Karoly: I think it was for military or government applications, and they refused to let me have it for this . . .

Misa: For this commercial application.

Dayan: This is the politics I've written about.

Karoly: Yes. I even got as far as; because the other people there; one of fairly senior software people said look, it would be a not very big project, possibly costing about $50,000 to write a microcode instruction set for that machine, to make it software compatible with the lower 3000 machines.

Misa: Would it make sense, Marcel, because you said George was the one who recruited you, maybe we should get that story.

Dayan: Alright. Just before I do that, let me tell you the story about Trevor Robinson, because obviously you are interested in the man. Tell you a particular story. When I was on my way back from the RIMFIRE study, which was in December 1968, I wasn't very happy because I'd bought some skis, I was intending to stay there, and all of a sudden, Jim McGeorge said the New South Wales TAB just put out an RFP, I need you back here immediately to write this over Christmas. Anyway, what could I do? So I came back; I spent the day in San Francisco; fly into Hawaii, had a couple of days in Hawaii; and then I was going to come straight back. As I was waiting to catch the plane in Hawaii, there was an absolutely gorgeous Pan Am hostess in front of me, and I started chatting to her. And she was traveling on the same plane to go to Sydney, because in those days, hostesses could travel for nothing. And I said, do you know anybody in Sydney? No, not really. What are you going to do? I don't know. The plane was going through Fiji and I sorta said, yeah, I wouldn't mind stopping off in Fiji and she said I wouldn't mind either. So I'm thinking, this is very good. So we started looking up flights. The trouble is the next flight from Qantas was two days later and I would have to explain how I got stuck in Fiji for two days. And just at that time, who should walk off a plane from Minneapolis, catching the same plane to Sydney, but Trevor Robinson. So I introduced Trevor to this girl and the first thing he did was to ask her whether she wanted to be his secretary. Anyway, nothing happened. And then when I came back I said to Trevor; Trevor, you know, I was all set to get off in Fiji with that girl. He said you stupid idiot, why didn't you?! [Laughter.] Well, this was in 1968. At the last Christmas lunch we had, which was the year before he died, which was probably 2003 and 2004, I sat down with him; the first thing he said was you're a stupid idiot not getting off that plane with that girl. Thirty-five years later he still remembered!

Misa: So he had that story connected to you. So he had an unusual people-centric memory.

Karoly: By the way, another story about Trevor, which is not that interesting; his wife was actually, he married one of my analysts.

Misa: Oh really?

Karoly: Yes. She was a very promising analyst. Out of the blue, I heard that they were engaged. In fact, I was in Brisbane on business and she rang me, did you know I got engaged with Trevor? I said no, I didn't. Good luck to you, or something.

Dayan: Very, very special man. He also told me an interesting story at that particular lunch. He was appointed by Senator John Button as an advisor to the Commonwealth Government, and one of the jobs that he was asked to do was to see how they could improve the use of computers through all the various government departments. So he did a study and gave it to Senator John Button, and apparently John Button called him in and said "Trevor, you've done something that nobody else has ever done. Every department is in agreement, they ***hate*** the report." [Laughter.]

Karoly: That was accurate.

Misa: So he hit some kind of nerve there.

Dayan: Yes, a very special man.

Karoly: I don't think there's much more I can tell you about when I led the crew. I got more directly involved with the TAB projects later, when Australia had its own systems division. Over to you, I think.

Dayan: Little bit of background. I was brought up in Tasmania and I did a degree in electrical engineering and in science; I did a double degree. And in those days, there was no digital computers at the University of Tasmania, there was only some analog computers, but I got quite interested in them. And after doing a little bit of work for the Hydro Electricity Commission doing high frequency radio tests, I headed off to Europe with the view to getting into what I called "control with computers." After being offered a job with the Space Research Centre in France and suddenly finding that it wasn't "control with computers," it was in fact to use a radio telescope in the South of France to measure high frequency radiation from the ionosphere, I decided no, I'll go to England. I got a job there with a company called International Systems Control, which was a fifty-fifty partnership between GEC and TRW [Thompson Ramo-Wooldridge Inc], and they used the TRW 330 computer for "Process Control." I'm not sure whether it was the TRW 330 or the TRW 130 that was used on Polaris submarines, but my experience was that the mean time between failure for these computers was about 24 hours. So I believed they never would've got a Polaris missile out because the chances were the computer wouldn't have been working! Well, I suppose it was the early transistors and capacitors that were just constantly going wrong with it.

Karoly: Even with Control Data computers, we had daily maintenance periods, but still, it was still a good day if we could manage a whole day without a failure.

Misa: So eight, or 10, or 12 hours would be good?

Karoly: Say 12 hours, maybe. You know, eight to 12 hours was the mean time between failures.

Dayan: They were incredibly unreliable.

Karoly: Well what do you expect from back then?

Dayan: Well that's right. Now the other thing you're talking about, programming, the TRW 330, you had to program the drum; there was actually no core memory. And what I remember was that an addition and subtraction was two words on the drum, but a multiplication was 35. So what you'd have to do was you'd fill up the drum and then you'd go back and look at where all your spaces were and then you'd fill up all the spaces with your instructions.

Karoly: Sounds a bit like the programming the G-15, which had a drum memory.

Dayan: Yes, that's what you had to do. And we didn't use punch cards, we used a teletype with punched tape. So every time you made a mistake you'd have to redo the tape again.

Karoly: So did the G-15. That was a tape machine.

Dayan: That was in 1964 and the project I was working on was with Courtaulds in Grimsby, and it was to help automate the production of rayon, or fibro, as it was called. My job was to keep the slurry, which is the wood pulp, with caustic soda [NaOH], at exactly 18.9 percent. So you had to keep monitoring it, and you had to either control the wood pulp going in or control the NaOH going in to keep it exactly at 18.9 percent.

Misa: So that's computer and controls, it was an application you worked on process control.

Dayan: Process Control, it was process control. And at that stage, I was actually thinking of doing a Diploma of Imperial College in Process Control [which was equivalent to a Masters Degree], but just at that time there was an Australian who visited us, who actually happened to be from Tasmania as well. I happened to have delivered the newspaper to his house many years previously when I was a paper boy. And he said to me, "I've got a very interesting project in Broken Hill, come and work for me." And I sort of jokingly said I will if you pay my fare back. Next thing, I know I get a letter offering me a job, offering to pay my fare back together with a very good salary plus in those days what they called, "the lead bonus." So this worked out to be a very, very good salary. And so after pondering how to finance the Diploma of Imperial College, compared to a good salary in Broken Hill, I decided to come back to Australia. And the job was quite interesting. The mine there is a silver/lead/zinc mine and the flotation process is where they crush the ore, and then they use xanthate and various other chemicals to create bubbles. And the idea is that the lead ore is heavier than the zinc ore, which sinks to the bottom and the zinc ore sticks to the bubbles. And the bubbles with the zinc coating are then allowed to overflow and this is the way they separate the lead and the zinc. We did a lot of regression analysis to find out which of the various reagents they were throwing in were actually working because it was an art, there was actually no science behind it.

Karoly: I can imagine.

Dayan: And we used the CDC 6400 at Adelaide University to do our regression analysis. So we'd go down there with a bunch of cards and spend all night with this thing crunching away. From our analysis we believed that we could save them a million dollars a year for each separation plant by using process control to handle the correct addition of the reagents. And there were two plants, so potentially $2 million saving. Peter McGregor was in fact the Control Data salesman who used to come round, and he was going to lease us a CDC 1700 for $100,000 for one year to prove that this saving could be achieved. And we put up a paper to the management of the company [CRA - Conzinc Riotinto Australia], at the time, saying look, we want to spend $100,000 to prove we can save two million a year. And the answer came back, "I don't understand computers, I've got better things to spend our capital on such as Hammersley [iron] and Bougainville [copper]."

Misa: So that could've been a huge money saver.

Dayan: Well, but that was the attitude. This is when I said I'm out of here. [Laughs.] And I said to Peter, are there any jobs going? He said I'll get George Karoly to come and interview you. So there's George, comes off the plane about 11 o'clock and I think we probably chatted for about an hour, then I took him round and showed him Broken Hill, and then we had a wine tasting and I poured him on the plane at about 10 o'clock at night. [Laughs.]

Karoly: That’s about right. I think you nearly got a speeding ticket.

Dayan: I did; and I got out of the speeding ticket when I got to Thailand and wrote to them and said come and get me. [Laughs.] The interesting thing is this policeman that wrote the ticket, I knew him personally and he was drunk. And I’m saying to him, “Merv, it’s me, Marcel!” That’s Broken Hill for you, yeah.

Misa: Okay. And just to connect those, that’s the same Broken Hill, that’s BHP site.

Dayan: Right, well that’s where BHP started. They started here.

Misa: So this is, so to say, at the center of something that became immense.

Dayan: Yes.

Karoly: Initially, wasn’t that a high, huge surface rock?

Dayan: Yes it was. There was a hill and there a guy called Charles Raspe suddenly found that it had lead and zinc; and he together with six others floated the BHP company. If you go to Broken Hill — there is about a half a mile wide and about two miles long hole right in the middle of the mountain, and they just took out all the ore and then left.

Misa: So, Marcel, you're really working for, you say, one of the computer customers but then experiencing frustration that they really didn't get it, and ended up coming to work for CDA.

Dayan: Yes, it was a mining company, CRA, which is of course is now Riotinto.

Karoly: The BHP later was a big Control Data customer. BHP Newcastle, well they bought a CDC computer for Newcastle.

Dayan: Yes, it's much later; this was 1967. So I started work in Sydney in February 1968, which is where I wanted to be, and working on some proposals on, again, Process Control; I think it was some proposals to control electricity using 1700s. I was also told to learn to work the 1700 at Sydney [University].

Misa: Can you describe the [CDC] 1700 a little bit?

Dayan: It's a 16-bit computer. I mean, well, what else can I say about it?

Karoly: It's not terribly good software.

Dayan: No.

Karoly: It had; well, they used assembly language. Did it have FORTRAN?

Dayan: No.

Karoly: No, and nothing much else, so it wasn't easy to use it in a system.

Dayan: It was the lowest in the Control Data range. So basically, there were three strings; it was the 16-bit range, which was the 1700; there was the 3000 series, which was the 3300, 3200, 3100 [interrupted]

Karoly: Actually, there was an ancestor of the 1700, except it was . . .

Dayan: The 160A.

Karoly: . . . which was also 16-bit, but it actually had the FORTRAN compiler.

Dayan: With the 160A.

Karoly: Yes.

Dayan: I never dealt with that.

Karoly: I'll tell you where I got involved with it. That might actually be an interesting tidbit for you. The Bureau of Census had an enormous amount of historical data on punch cards, and they wanted them transferred to magnetic tape. Control Data, I don't know what the commercial arrangements were, probably leased them a 160A, which was a 16-bit machine with certainly the FORTRAN compiler, don't think it had an operating system. And actually it was Leonard Whitehouse who wrote the program.

Dayan: This was before my time.

Karoly: Len, a Control Data employee, wrote a program for it to convert the card data to magnetic tape.

Misa: And they could've had hundreds of thousands of these cards, or maybe millions, even.

Karoly: Oh yes.

Misa: A stupendous number.

Karoly: Ron Bird might have some figures on it because he was around, as also would John O'Neil. And one of them might even know how to contact Len, because Len should still be around. He would be, I'd take a quick guess, he'd be about 50 now, he was quite young.

Dayan: Oh, George, come on.

Karoly: Sixty maybe.

Dayan: Come on, come on, come on. And the rest. Come on, he was older than me and I'm 71. Okay, of course he was.

Karoly: Could be. I thought he was younger.

Dayan: No, no, he must be in his seventies.

Karoly: He might be in his seventies but he might still be around. He was at Monash University for a while. If I can be slightly rude about it, the program he wrote wasn't terribly good. The problem was he had no sort of practical experience. I mean, he could write good FORTRAN programs but he didn't realize one of the important things was to write a program which coped with things like misfeeding cards, dirty cards, jamming cards, and they had all sorts of trouble with that.

Misa: So they needed to deal with mistakes or the exceptional cases.

Karoly: Yes. Some of these boxes of cards, they actually had to vacuum cobwebs off before they could feed them into the card reader. [Laughs.]

Dayan: Getting back to your question, there were three lines that Control Data had: a System 17; the 3000 series, which had a 24-bit word; and then the Cyber series which had a 60-bit word, didn't it?

Karoly: Actually, the 3600 had a [interrupted]

Dayan: Okay. That was a little bit different. It wasn't the same as the 31, 32, 3300.

Karoly: No, that was after; so it was the 34 and 36.

Dayan: That was part of the problem, there were so many computer lines and there was no commonality between them. They all had different operating systems, they had different languages . . .

Karoly: In a way, the 34 and 3600s were regarded by Control Data as their principal large computers.

Dayan: Yes, before the 6000 and Cyber range came along. But that was designed completely separately, this was what's-his-name?

Karoly: Cray. Seymour Cray.

Dayan: Yes, Cray. The Cybers, whereas the others were designed by a different team and that's why never the twain shall meet.

Karoly: I have a feeling Cray had something to do with the original, the father of the 3600, whatever it was called. [I think it may have been called 1604 – someone in the USA may know. GK]

Dayan: I think that had a long word as well, before the 3600. Yes.

Misa: The Cybers were, in some way, my understanding is that they were a successor to the 6600.

Karoly: Vice versa.

Dayan: Vice versa, yes.

Karoly: Cyber was the development machine and 6600 was the commercial machine.

Misa: The commercial machine, okay.

Karoly: Commercially produced.

Dayan: And then they produced the 6400. The 6400 was a cheaper version of the 6600.

Karoly: Yes, you're right. So the Cyber was, in fact, some way — I should have brought it because . . .

Dayan: Actually, it wasn't the Cyber; the 6600/6400 were the ones that . . .

Karoly: Sorry, I never thought to bring it; I've got a numbered copy of an engineering manual for the Cyber.

Misa: That'd be something to hang onto, George.

Dayan: I'm sure you've got access to a different type of stuff. So anyway, here I was; started work in Sydney. And then I think it was you, George, that asked me if I had any moral objections to working on a horse racing project because of course, Control Data in those days, was a very moral company. And the second question was "did I want to go to Minneapolis for about three months." And I remember saying what's the catch? And the catch was, you've got to come back to live in Melbourne. There was all this Melbourne/Sydney rivalry, you see. And being single, I said I could handle that. The project, which was called the RIMFIRE project, was following the success of CARBINE, which of course you know Ron Bird and the guys will tell you all about. [Carbine was a famous racehorse that won the Melbourne Cup in in 1890. The system was designated **C**omputer **A**utomated **R**eal-time **B**etting **I**nformation **NE**twork. Rimfire was descendant of Carbine and won the 1948 Melbourne Cup at odds of 80 to 1. The system was designated **R**emote **I**nput **M**achine **F**or **I**nvestments on **R**acing **E**vents.]

Karoly: It was an off-course totalizator.

Dayan: Well, CARBINE was basically a way of computerizing the telephone betting side of the betting and consolidating the agency totals

Karoly: As well as; yes, you're right. It does do that.

Karoly: That'll give you a good lineup of the history. That's why I brought it.

Misa: Thank you.

Dayan: Do you understand totalizators . . . ?

Misa: Enough to have you describe them. I don't know the details but if this is one of these problems that actually occurs several times in computing, it's quite fascinating. Go ahead.

Dayan: Well, let me give you a little bit of the history of totalizators, because for many, many years, only on-course betting was allowed in Australia and New Zealand. So in other words, people who wanted to have a bet on a horse had to go to the racetrack. And it was the same in the U.S. It wasn't the same, interestingly enough, in other countries. My parents were living in Egypt before the war and my father was, would you believe, employed to sell bets outside of the racecourse and one of the things he did was to go around the cafes and actually take bets. Then they brought them in to the race course and consolidated them with the on-course bets. The whole idea of a totalizator is that you take the money in for a particular race and bet type [called the pool]. Let's say it's $100,000 is bet on the winner of a race; you then take your share out of it, which is, say, $20,000; and then the $80,000 that's left is distributed to the bets on the winning horse. So if there is 8,000 tickets on the winning horse, then everyone will get $10.

Misa: That's the winning share.

Dayan: Yes. But if there is only 4,000 tickets on the winning horse, everyone will get $20. That's the whole idea of a totalizator. So the totalizator company can't lose money because they take their share off the top and then distribute the rest, so it's a very nice business to have.

Karoly: And in fact, from that summary you could work out that the aim of a totalizator company is to get the maximum number of bets because they just take a percentage of the money. They don't care what horse wins, but the more bets there are, the more money they make.

Misa: The greater amount of money coming in, their 20 percent gross, regardless of who’s winning or not.

Karoly: They couldn't care less.

Dayan: After the Second World War, there was an explosion of what was called SP, starting price bookies that used to basically have their own area in each of the pubs and illegally taking off-course bets. It was actually in New Zealand, a guy called Morrie Smythe, who first said to the government, look, you’re really missing out on a huge potential revenue. All this illegal stuff going on, you could in fact, legalize it and tax it. So he would have been a major or similar rank in the war, and he had very good organizational skills. He developed a system whereby you would hand write a ticket, and there would be three copies of the ticket, an original and two carbon copies The top [original] copy would be given to the punter, the second was what they called the collation copy, and third which was called an audit copy, which would be kept in a bottom drawer, in case there was a hint of fraud or anything untoward. And this collation copy, what would happen is, as soon as the race closed, which was usually half an hour or an hour before the actual race took place; these collation copies would be used to count how much had been bet on horse number one, how much on horse number two, and so on. And in fact what Morrie Smythe did was he developed a plastic tray that had 25 little pockets [5x5], one for each horse, because there was 24 horses, and one was for any horse that was scratched [i.e. a horse not in the race]. And so by setting it up in these pockets, you could very, very quickly count up how much had been bet on each horse.

And that what would happen is 30 minutes before, each agency that sold these bets would call in to a central point and say I've got 26 on horse one, 35 on horse two, and so on. And at the central point, the person would sit there and would write it all down, you see, and . . .

Misa: And each bet would be the same amount of money?

Karoly: No, they were in units.

Dayan: Actually, they were in 50 cent units, but it would say this is how much you've got. Someone at the head office would say, okay, this particular agency in whoop-whoop has got the following. And then another agency has got the following. So they would write all these numbers down and then they would add them all up. Then you know how much you've got in total on horse one, how much you've got on horse two, and so on. And then you add it all up and know how much you have on the whole race, and then they would sit down and work it all out, take off the 20 percent, and say this is the amount of money we've got to actually give out. And then when the horse, whichever horse won, would then say okay, this is the dividend. They would then advise each of the agencies, this is the dividend. Then when the person came in with the ticket, they would then match it with their collation copy to make sure it hadn't been altered, and then pay it out.

Misa: Okay. Was the payout done centrally or in the remote?

Dayan: No, in that particular branch or agency, actually. Mostly it was agencies because they would set them up — this is why they're called Totalisator Agency Board — they were individuals who were contracted to provide the service. And the premises were usually owned by the TAB, and they had a contract with the agent based on the percentage of turnover, and a fixed rate. The agents would employ casuals because most of the races are run on Saturday and so allows the casuals to earn extra money. So you can imagine the collation process of sitting down, adding all these numbers together, first you had to write them down, then someone had to add them, and someone had to check them and then check them again. The original CARBINE provided a keyset in each of these agencies which allowed them to key in the totals on each horse and transmit this to the central computer. I'm right George, aren't I?

Karoly: I think so. By the way, hadn't somebody in New Zealand already done a similar system?

Dayan: No. So, basically, the CARBINE 1 used a Plessey keyset . . .

Misa: Plessey keyset?

Dayan: . . . Plessey, from an English company called Plessey.

Karoly: Made here, though.

Dayan: And basically what would happen was once all of these tickets had been manually counted, they would then transmit that information to . . .

Karoly: The head office.

Dayan: . . . the CARBINE computer and that's what CARBINE did. It added up all the so-called investments, they're not obviously investments, but all the bets on horse number one, on horse number two, and so on, for the Melbourne race three, and then Sydney race four, and so on. There'd be up to about a hundred races on Saturday.

Karoly: And of course, there would be more than simple bets, there would be more than one bet, you could . . .

Dayan: Well you had win, you had place, you had quinella, and you had a double. So you can see that . . .

Karoly: A lot of computation, a lot of arithmetic to be done.

Dayan: It wasn't complex computation but there was a helluva lot of numbers that you had to add up and . . .

Karoly: And get them right in a fairly limited time.

Misa: That’s right.

Dayan: That was the other point, was that people would want to bet . . .

Karoly: People would want to bet as close as possible to the start of the race so they tried to minimize the time that TAB had to add all of these things up.

Dayan: See, that’s the whole point; it used to be that the betting closed half an hour before race start time. And the reason that not only that the central authority, in the case TAB, had to actually work out the number, but then they have to actually transmit that information to the racecourse, There it’s amalgamated with all the bets taken on the racecourse because it’s the racecourse itself that determines the dividend using the total of the off-course betting and the on-course betting. So, it’s as I said, there’s nothing really magic involved in the calculation, it’s just the sheer volume of numbers and trying to do it quickly. So when CARBINE was developed, which included bringing all the stuff in from the agencies, and also telephone betting. Telephone betting is where people set up an account with the TAB, and they would deposit the money through the agency, and they would then be able to ring up and actually place their bets by telephone.

Misa: From somewhere else.

Dayan: From home, usually. And so by placing the bet by phone, again, in the old days it was all done by hand. You know, they would write down the bet details by hand and this copy would go to a central location to be added up; so CARBINE automated the telephone betting and brought in all the consolidated bets that again had been sold manually in the agencies, into the central system. And there's lots that's been written about it. Ron Bird has written all about it on the Ex-CDA web site. It was one of the first fully backed up real-time systems, and it was a very big success. So the next stage was why don't we actually automate the selling and paying part with a terminal?

Misa: And that was RIMFIRE?

Karoly: Yes, that was RIMFIRE.

Dayan: And so what they worked out; the best way to do this was to set up a joint team made up of Control Data people and TAB people, and of course, there were some advantages to having all these people go to Minneapolis on expenses; [laughs]; so I joined . . .

Misa: So this group is the CDA people or did TAB people go to Minneapolis?

Dayan: I was the only CDA guy. There were three guys from CDC came here, and the four of us were then put through a month of understanding how the manual TAB system worked. First of all, we had to go though an "agents" course conducted by the normal guys that used to teach agents. This was a full week's work and what was interesting, by the end of that week, was that they said this was by far the best group they'd ever had because up until then, most of these agents were in fact, ex-bookies. And the TAB in those days was such that they were told you will do this, you will do this, you will do this. Of course we kept saying why? Why? And now all of a sudden they had to think why are we doing this? And it was actually very good because there was a lot behind all of their procedures and it actually made them think.

Misa: Because you were trying to understand the structure of the system.

Dayan: Exactly, and we were saying why do you do this? We actually had to get on the phone and listen in, and take bets ourselves. So for a whole month we were totally immersed in the manual system. And then we went to Minneapolis and we were locked up in a room. There were ten of us, there were six people from Victoria TAB, there were three CDC guys, and myself.

Misa: So three CDC guys from Minneapolis.

Dayan: Yes. There was a sort of a manager guy, called Bill Criego; there was an engineer, Milt Spieler, and there was Orrin Butterfield, who was a guy who had communications expertise. And we basically tapped into all the resources of CDC. Well, the trouble was that there were a few personality clashes. The guy who was running the TAB team at the time [Charles Scorgie] was a very, very strong personality, and unfortunately, Bill Criego was just well, young would be the best way of putting it. And Charles Scorgie dominated him. So they brought in John Miller, who was a much more senior guy to bring Scorgie under control. Which he did, because for a while there the study was going off the track, and out of control. The actual problems were pretty major because for a start there was no selling terminal.

Karoly: No suitable terminal.

Dayan: No suitable terminal. What we found was that there was a printer that was being used in New York, in a theatre application. Control Data was actually selling theatre tickets. Imagine a drum printer that actually had a drum that printed the tickets. We had to design the electronics, we had to design the packaging and keyboard, the whole works. Plus, of course, you then had the communications, and comms were bloody expensive in Australia and very slow; 150 baud; 150 bits per second.

Karoly: Occasionally you could get 240 [bits per second], if you were lucky.

Dayan: Well, yes, but remember that we actually put two RIOT [Remote Input Output Terminal, that's what the terminals ended up being called] on each line, so one would be transmitting while the other would be receiving, just to get the speed of this thing going. So essentially, the basic design was done and the group came back after about three months; and Charles Scorgie got fired after he got back [laughs] because of some of the shenanigans that had gone on.

Misa: It was one month of manual training, but you were in Minneapolis three months then?

Dayan: We had one month here, which was the whole of May; and then we got to Minneapolis mid June; and I stayed there until mid December. But the rest of the group…

Misa: From June until December?

Dayan: December. I was there for six months. But the group actually came back after just about three months, the TAB group; and I stayed on to keep the project going.

Karoly: By the way, what was Scorgie's boss' name?

Dayan: Ken Davis.

Karoly: Did you know that I nearly became a consultant for Ken?

Dayan: I didn't know. There's a picture of the group in Minneapolis.

Misa: Oh, okay. Bill Criego, that's C-R-I-E-G-O.

Dayan: I'll give you this. This is basically the write-up.

Misa: This says RIMFIRE design team. That's great.

Karoly: And this is just an outline of the whole history. Actually, CARBINE 1, which was just purely the central totalizator; CARBINE 2, which expanded and improved the telephone betting system; then RIMFIRE, which automated the ticket selling and automatic collating.

Dayan: What happened then was that the TAB themselves had a lot of problems because Charles Scorgie was fired, and Bart Goodwin took over. As I said, in December I was asked to come back here because of the New South Wales TAB, as it seemed that the Victoria TAB was doing so well, NSW TAB decided they would automate as well and so they put out an RFP. So we worked on this RFP. Unfortunately IBM had a much closer relationship with New South Wales TAB and we didn't get the contract. In fact, IBM had initial problems with the system implementation but eventually got there, but it was nowhere near as good as the system we had. The other thing that happened . . .
then was that CARBINE had been going so well, especially the telephone betting [because you could now bet much closer to the time], had absolutely blossomed and they needed to have extra capacity [i.e. to handle more telephone betting accounts]. So at that time, it was based on 3100s with only 32k of memory. And so the idea was to replace the 3100s with 3300s and double the amount of memory from 32k to 64k, except it was in banks so when you addressed the second bank, you couldn't just flip from one to the other, you had to actually know which bank you were addressing. So to actually convert

the code was not a straightforward exercise, not a trivial exercise at all. And when I came back I was put on that team to work on with Steve McCusker, Mike Clayton, Kurt Imberger, and Alister Dowling, and Ian Wadham was basically trying to run things. So I then spent basically a year working on this, and the idea was also to replace the Plessey betting keyset with a CRT based telephone betting terminal, from a Canberra Company called Information Electronics.

Dayan: Anyway, so it was a separate project; it was called CARBINE 2, and again, you'll probably see it written up on the Ex-CDA web site; but the idea was to, if you like, expand the number of accounts the system can handle, provide a better telephone betting terminal and also to make provision for linking it to RIMFIRE.

Misa: So RIMFIRE was still under discussion?

Dayan: Well, it was designed; it was basically a design and there was a lot of too-ing and fro-ing about the price and the cost and the timetable. Now, the interesting thing is that eventually the contract was signed and it was for $6 million to develop the RIMFIRE system with was it 1000 or 1100 terminals? But I remember the terminal was priced at $3,800. Now, there's a suburb where my wife's parents had a house, and at that time, that house was worth $12,000. So it was one-third the price of a house. And today, that house is worth about a million dollars. [Laughs.] So that puts it in perspective. That terminal today is probably about $50, you know? How the hell they were able to justify spending

$3,800 on a terminal in those days; but it just shows you how profitable the TAB was in those days.

Dayan: I'm actually just quoting here. The ticket selling machine each worth $3,800 will be manufactured to TAB specification. This is how Control Data was able to set up a manufacturing plant in Australia to produce these terminals, the famous RIOT [Remote Input Output Terminal] terminals as they came to be known. And, of course, that's all covered by John O'Neil and the establishment of the Australian Systems Division. This RIMFIRE project was, if you like, what gave Control Data the base which allowed the Australian Systems Division to be set up, and which allowed the manufacturing to be set up.

Misa: Because you wouldn't need ten terminals, you needed how many?

Dayan: Oh, I think the initial number was 1000 and was eventually expanded to 1100.

Misa: 1100.

Dayan: Actually it eventually expanded to 1000 RIOT's and 120 D-RIOTS [Dual RIOT] which gave them 1240 selling windows. The initial RIMFIRE contract was five 1700 computers, and 1,000 RIOT betting terminals; to a total value of around $6 million.

Misa: The terminals themselves, that's almost $4 million right there.

Karoly: Yes, 1000 RIOTs plus 120 Dual-RIOTs later.

Dayan: What was interesting also was this was the price for the hardware only, and the software was thrown in for free. In 1993, I had my own company with a couple other guys, and we developed a system for virtual horse-racing, and it ran on PCs. And we could afford to throw in the PCs for nothing and just charge for the software.
[Laughter.]

Karoly: How the world changes. Of course those PCs were probably a helluva lot more powerful than those 1700s.

Dayan: I'll tell you another interesting little story attached to this. I was asked in January of 1970 to become the project manager for RIMFIRE. I was never all that keen on programming; I did it but I always preferred being involved with people. And I knew nothing about project management, but obviously, I had a good relationship with the TAB because of the previous study and everything else.

Misa: That was in 1970?

Dayan: January of 1970, and I went over to Minneapolis with the TAB to see how things were going with RIMFIRE because by that stage the project had already started. And the whole idea was that the first 50 terminals would be manufactured in Minneapolis together with the software which was also being developed in Minneapolis, and then in August of 1970, the software team was all going to come back to Melbourne with the 50 RIOT terminals and then finish it all off. It had to be running by May of 1971. So I went over there to see how things were going; and then we went back there again in April; and then there was the final trip in July/August, where the TAB were going to take delivery of their 50 terminals and at the same time they were going to go through and test the software, so August was the big date. Now, it so happened that a new entity called New York OTB [Off-Track Betting] put out an RFP in early 1970 to bring in off-track betting to New York.

Misa: New York Off Track Betting is something similar.

Dayan: New York Off Track Betting put out an RFP. And immediately Control Data in Minneapolis said hey, come and help us write a proposal.

Dayan: We had the big acceptance tests scheduled for August and just before that, well, when I got there to Plymouth, they said to me oh, sorry, you've been bumped. I said what do you mean I've been bumped? There's a thing called a DART terminal that's taken priority and you're not going to get any RIOT terminals in August. What am I going to do?! The TAB guy is coming over and they said oh, that's tough. Of course, you know, being an Australian you can't monitor these things that close up, obviously the local guy's in a better position. Anyway, I had Bart Godwin there and I said, look, we've got a bit of a problem. How many terminals would you be prepared to accept? I also went to the manufacturing plant and asked if you were to stop making DARTs today and started making RIOTs, how many could you make by the end of August. He said we could probably make 20. I went back to Bart and I said would you accept 20 terminals. Sorry, it was near the end of July. And he said, well, if that's the way it is, it has to be; yes, I'll accept 20 terminals.

Misa: You're doing proof of concept . . .

Dayan: Yes, the basic software was there; you're going to run tests on it, and volume tests and all the usual type of stress testing. The software was actually not in bad shape, but you needed terminals to be able to do anything. Now, the proposals for the Off-Track Betting System were in to New York City, and we had to go and do our presentations to the NYOTB evaluation team, and it was in July. I remember that a whole group of us went to New York, and there were all these deputy mayors there asking questions, and I always remember one question from a black guy who got up and he said "I don't see any black faces." Oh my God. My boss in Minneapolis was black; his boss was black; we had at least two customer engineers with us who were black but we didn't think of putting them up front.

Misa: As a part of this travel . . .

Dayan: Well, the group that were doing the presentations were all white. Shock, horror but one of the Control Data VP's answered " …but we have manufacturing in Israel." [Laughs.]

Misa: Not the same constituency.

Dayan: But, I mean, that was; unfortunately, they were a little bit out of their depth and AMTOTE, who'd been running all the racetracks in New York were obviously in a much better position. But that night we stayed in New York, and the guy who was in charge of the sales effort, Tom Moore had actually had been a politician, and he knew Bill Norris personally. And I went to him and I said, you know we're not going to have any chance of getting this contract, and he said why not? I said look, we've got the Victoria TAB that this whole system is based on, and your group from New York, after these presentations is doing a tour around the world, and on August 8 and 9, they're going to be visiting the Victoria TAB. And, you know we've missed all our deadlines, they're going to be totally pissed off. He said, what can I do about it? I said, it's all dependent on these RIOT terminals. He said I'm going complain to Norris immediately. And I tell you, that manufacturing plant went CLUNK, stopped producing DART terminals and started producing RIOTs.

Misa: The terminal you needed in quantity.

Dayan: Yes. I was amazed. It just shows, you know, if you're at the right place at the right time, you could have influence.

Karoly: Also, in the United States, you'd better do it.

Misa: That's right. If Bill Norris says A, B, or C type . . .

Karoly: It's going to start; they start making that RIOT type the next morning.

Misa: Or earlier.

[Laughter.]

Dayan: We got accepted, although there were a lot of problems. The group came back to Melbourne, completed the software, and it was due to go live on May the first. It went live on May the third, which was in fact the first Monday after May the first. It was actually done on time. I think it was done within budget but I wouldn't swear to that, but I think it was because there was a lot of fat in that budget, as you can imagine.

Misa: To do development, you want to have an ample budget.

Dayan: Yes. When you think about the fat there was in things like that. And of course, it was the basis of the Australian Systems Division; so that was the RIMFIRE project.

Karoly: Next thing, I suppose you can talk about the New Zealand TAB.

Misa: New Zealand TAB.

Dayan: New Zealand, yes. What happened then was because of course, RIMFIRE was very successful and went live on time. This was on May the third, 1971. At about this time, New Zealand TAB hired some consultants [PA Management] and issued an RPP to computerize their operation. And naturally enough, I was involved in doing all the technical backup work, and Peter Dulmanis was the salesman.

Karoly: I remember going to New Zealand with you over something, but I actually don't know what it was.

Dayan: Well I'm not sure at that particular time; well there were lots of different times. I remember that the first time I went over there was in August 1971, because I tried to go skiing and there was no snow, and then the proposals had to go in. Well, when we really knew that we had won the contract was when we arranged for a telephone link back to Australia and the stand-by RIMFIRE module, and we actually ran a phantom race meeting. I'd got some videotapes of some races in Auckland, and we took a RIOT terminal over to New Zealand, and we actually sold bets on the race and processed them in Melbourne and then paid them out. Basically just ran the whole thing and just showed that we could actually do it.

Misa: Processed, real time, with a good race, it gave a sense of does it work?

Dayan: Yes, the whole thing. And the funny part was that we gave a bottle of whiskey to whoever would win the most monopoly money on the race. And it was the TAB consultant that ended up winning the whiskey, and somehow or other, they were not happy that the news got out into the newspapers. [Laughs.] But we had a lot of competition from IBM because Control Data was not set up in New Zealand and of course, IBM and ICL were very well established there. And IBM did everything they could to stop us, you can imagine; they went at the political level, they went at the board level, they went at the executive level; it was a very, very tough sell.

Misa: ICL and IBM had already established subsidiaries in New Zealand?

Dayan: Yes.

Misa: But curiously; I'm not saying that CDA ought to have, but it's pretty close.

Dayan: In fact, the only thing CDA did prior in New Zealand was to sell a message switch to the Ministry of Defence. So they did have one setup, but there was no office [just a representative called Flight and Field Services run by Digger Harding], there was nothing else, it was all done out of Australia. Ministry of Defence; because they had sold the 1700 here as a message switch to the Australian Ministry of Defence, and because there was such close contact between the two Defence Departments, it was logical that they bought the same thing. So there was no real competition on that whereas this particular contract was very, very hard fought. In fact, one of the interesting things is that ICL approached us and said why don't we go in this together? We've got the base, we've got organization and maintenance, and you've got the knowhow. And so we said okay, well let's explore this. You're familiar with ICL aren't you, the English company here?

Misa: Yes.

Dayan: Finally it got down to, at a meeting, was how much are you going to charge for your computer? And they puffed out their chest and they said whatever their cost was; and we then said this is how much we're going to charge, and it was a hell of a lot less than they were. And they said how do you do that with a 3300? And we then said no, we're going to do it with 3100. And they said how can you achieve that processing power with a 3100? And I said, that's our knowhow. [Laughter.] That was pretty good. So anyway, we decided we'd do it on our own and this is why this is very, very personal. I had a girlfriend in New Zealand, and I had a girlfriend here, and then all of a sudden, in December, it looked like this was all going to happen. And they insisted that we do the project over there so that meant I had to go over there and live there for two years.

Misa: So that was December of 1971; so this is all happening in that launch year of 1971?

Dayan: Yes. And I had to go over there and put together a team and take that team over there. And so I got the hard word put on me and I got married in February of 1972; and three other members of the team also got married. One got married in July, Bob Jensen; Phil Stokes got married; and so did Stuart Broad. There were four of us that ended up getting married just before we went over there. [Laughs.] And it was a very, very good group, as you can imagine; they're all very, very good programmers.

Misa: How close was the New Zealand TAB thing to what you'd already done? I know it's not just doing a duplicate, but were there different ways the New Zealand TAB was organized that meant that the system needed to be adapted?

Dayan: Yes, and I'll explain to you about this. I should also tell you that the New Zealand TAB was the first TAB to be set up in the early 1950's; and then the Victoria TAB was set up in 1960; and seven people from New Zealand came over to set it up so there was a lot of links backwards and forwards between the two. They were very, very close; so they knew everything that went on. One of the problems with RIMFIRE was that there was no unique serial number on the ticket, so what was in fact happening was that there were a few scams whereby the agents themselves, after they had got the tickets in, would go to another agency and cash them again, because there was no unique serial number.

Karoly: Winning tickets.

Dayan: A winning ticket.

Misa: Oh, these winning tickets collecting up and down the street.

Dayan: Exactly.

Misa: So there's no way of saying the winning ticket has been called in.

Dayan: That's right.

Karoly: All the system knew was that they were winning tickets. They didn't know what the numbers were.

Dayan: That's right; they didn't know the numbers. It all went back to the fact that there was so little memory storage. There was so little storage you couldn't have a unique serial number. Today, it's obvious; you have a unique serial number; you mark it off; you couldn't do it in those days. So in actual fact the way the Victoria TAB solved the problem is that as each ticket did have a unique identification number, you could find out if the ticket had been cashed twice, but this could only be worked that after the fact. So what they did is because every transaction ended up on that magnetic tape, they then wrote a program that subsequently would go through every transaction, every ticket, and find how many duplicate payouts there were. And then they could work out a pattern that there was a particular agency that would cash these winning tickets, and two days later, in some other part of Melbourne, some of those tickets would be cashed again. So they built up a pattern of who was doing this, and then they swooped and got all these guys and then charged them with fraud and stealing and made scapegoats out of them and gave them fines and jail sentences. And that stopped it, especially as none of the agents and casuals knew how they had been caught.

Misa: What they were doing was illegal, right?

Dayan: Of course. It was an absolute scam. So to get over this problem, the way that we solved it was we ended up putting a bitmap on the 3100 to represent each ticket sold on the 1700's. Again, because memory was so limited; and we had a bitmap for every single ticket sold representing the serial number on the ticket.

Karoly: But it was in bit form.

Dayan: And we didn't actually keep the serial number, but we kept a bitmap of every ticket. So when the ticket came in to be cashed, we crossed out that one bit. And that was held in the central computer, the 3100.

Misa: So if somebody tried to take that same ticket, and cash it, was there some way then that the agent would say is this a valid ticket?

Dayan: Well, they just put it through the system and the system would then reject it, because it had that one bit crossed off. So that was one of the changes that we had to make to the CARBINE/RIMFIRE system to meet the NZTAB requirements. The second change — and that was the basis of GWS, actually — because the original CARBINE system only catered for four bet types; win, place, each way, and quinella.

Karoly: No, but it didn't.

Dayan: . . . each way was just win and a place, quinella, and doubles. Sorry.

Karoly: Yes.

Dayan: New Zealand wanted a treble, and the problem was that throughout the system there was only two bits that were used for identifying the bet.

Dayan: If you've only got two bits, it's only four bets that can be identified. And that was right through the whole system. It was two bits.

Misa: If you had three bits, then you could've had eight but two bits is four.

Dayan: That's right. So when they said they wanted to add a treble, I thought oh God, here we go. And I remember that Jim Walters, he was project manager on the original CARBINE 1, in one of my talks to him — and I can't remember when it was — he said he had a way, a generalized way of handling a bet type so that you wouldn't specifically say it was a win, or a place, etcetera. He called it CVS, it was a generalized way of handling bets. In that way, he said, you would not be locked in to four, or five, or six, you could have any bet type. I thought that's a pretty good idea, so I took that to the New Zealand TAB and said look, this guy's got a fairly good idea. Why don't we have him come out and see whether in fact implement this CVS system, which would then allow you to go beyond trebles. They agreed to pay for the study and so I invited Jim to come and do the study. So he comes out and he does the study, and blow me down, instead of keeping — and by the way, you could imagine, all we're doing is taking the existing system and modifying it — which you knew exactly what you had to do, and we had worked it out to be 149 man months, which was still a fair bit to do because we had to do new reports and all that. Jim does a study and it's completely different; it's all based on System 17s with all these TICS and TOCS all talking to each other. And blow me down, at the end of the report he claims it's going to take 149 man months to implement! And that night, which was at Digger Harding's place, and I remember he was pouring all these cocktails; and I said you're a bastard, Jim. I said, you've got two options, either you change the conclusion that says great idea but we just don't have the time to do it, or you can piss off and I'll do it. He said okay, I'll change it. And that night I said, you're a bastard; what would've happened if I'd believed you. He said it would've been your problem, wouldn't it? [Laughter.] So then he goes on to Australia with this TIC/TOC report, he goes in and sees Herb Hughes and Ray Sharpe, and he says look, I've got this fantastic system, called GWS.

Karoly: And he sells the software concept to them.

Dayan: And he calls it Generalised Wagering System. It's the greatest thing since sliced bread. TICs and TOCS, they're all talking to each other and there's redundancy and God knows what. And by this stage I'd locked up all the people that knew anything about TABs and totalizators in New Zealand. These guys go out and sell it to the Victoria TAB as a replacement for CARBINE and RIMFIRE. Not only that, but instead of writing a proper functional specification, what happens is Jim's report, basically a blue sky document, ends up as the functional specification, which it isn't. You could do this, you could do that, you could do that; and all of a sudden, there's a contract with the Victoria TAB for the GWS system committing to all this, which was pie in the sky.

Karoly: Completely, and it never worked properly on site. That's what I was talking to you about; it was trying to design software that nobody had ever designed before. Multiprocessing software, you know, interconnecting processors.

Dayan: These System 17s were very underpowered machines. And you had a whole bunch of them; these TICs and TOCs of transaction-oriented processes and a terminal . . .

Karoly: It was a system of; just trying to think of how many processors — Marcel can probably tell you more details —but it was quite a large number of processors dealing with the same data, designing it from scratch.

Misa: That was one of my questions, whether there were any other models that people could draw on.

Karoly: Not that I know of.

Misa: In terms of distributed computing, it’s hard to imagine what the model would be.

Karoly: Yes. The idea that there were smaller processors that process the incoming data, and then the incoming messages, if you like, you know, the bets. Actually, those processors were called TICs [Terminal Input Controllers], I think. And they’re the ones which took these bets and calculated totals, and all sorts of things in real time. And all of this, in a way that hardware or software failures wouldn’t affect the final result.

Misa: Because if you’re running census data and something has a hardware fault, you just run it again, but you can’t rerun a horse race.

Karoly: Census data don’t cost money, not really, and here you’re tossing millions around. To give you an order of magnitude — I don’t know what it is now — but I had something to do with Hong Kong TAB many years ago and they told me that their turnover there in the last hour before the race is several million Hong Kong dollars.

Misa: Just in the last hour.

Karoly: Yes. So you're talking about real dough.

Misa: Definitely.

Karoly: You don't want too many mistakes or too many fraudulent actions.

Misa: So the whole thing also has to be secure, again, because somebody tampering with the data could….

Karoly: Security and recoverability. But as far as I know, they never, ever had to use this Audit Tape Recovery. We kept what we called an audit tape on all these systems, which recorded every transaction in detail. The theory being that given that, it was possible to recreate everything, but you can imagine how long it would've taken, right? Forever.

Misa: An immense amount of time.

Karoly: It never happened, to my knowledge.

Misa: That was a kind of security that kept the system whole.

Karoly: Everything was recorded in duplicate on different pieces of hardware. Well, so it was a difficult project and there was some design errors made, no question about it. Design errors are human. In fact, rather interestingly, very late in the piece, one of the designers came up with an idea which would have simplified a lot of things, but it was too late. You see, one of the problems is that to attract more customers, TABs invent all sorts of fancy bet types and some of those . . .

Misa: Oh, of course, not just the standard ones that Marcel is describing.

Karoly: Some of those bet types have an incredible number of permutations so that storing them in the system gets to be a problem. Just take a fairly simple bet type, a double. It's simple, you pick the winner of two races; you know, horse 1 will win the first race and horse 17 will win the second race. But some of them are offering multiple doubles; you could say I want to bet that either one or 17 wins, or one and 4 wins, or two and 7, you know?

Misa: Yes.

Karoly: In fact, in the extreme, I think it was possible to have; but I don't think everybody that is betting on it, to have every possible combination of horses as a double, so you can imagine how many records are in there on the drum, in that record, you know.

Misa: Stupendous requirements to keep track of those very few types.

Karoly: So it was a challenge to plan these bet types. I can't remember the details of some of them, but there're all sorts of ways, weird things designed by salesmen to attract…

Misa: Bettors.

Karoly: Yes. But just causing some of the technical problems of designing systems like the GWS [Generalised Wagering System].

Dayan: Oh boy.

Karoly: Terminal Input Controllers, I think it was.

Misa: Terminal Input Controllers, if that's TIC, then TOC is . . . ?

Karoly: Transaction something. I can't remember.

Dayan: The trouble was that because everyone talks to everybody else, they spend all the time talking to each other saying you got this, you got that, and they don't have any time to do any processing.

Karoly: Oh yes, and there's no defined. I mean, for instance, one of the things that always bugged me, when a TIC received a bet and verified it, which TOC it sent to was a completely random event.

Dayan: Yes, that's right. It was all part of redundancy, you see.

Misa: Oh, okay. Just a way of talking about security and so this is one way of getting redundancy?

Dayan: You send it to two of them, didn't it? I'm pretty sure it sent it to two of them.

Karoly: Maybe, but at random, so there was no way of controlling the loading or anything, it was done at random. Not really at random, but ask Jim. [Laughs.]

Dayan: The New Zealand TAB system, again, very lucky, we managed to get it done on budget and on time. That was interesting because the group I had, we had a fairly good deal because there was no policy for Australia sending people overseas. So we used the Minneapolis one, which had tax equalization, and paid for your accommodation, all that, so we were on a very, very good wicket, you can imagine. So I had a very happy team. And at the end of it, I remember negotiating because all the salesmen were getting all these real freebies and things, and I kept on saying well, if we get this done ahead of schedule we should be able to get some reward. So I finally got approval to run a postmortem in Fiji and take the whole team and wives.

Misa: In Fiji?

Dayan: Fiji. And so we did; in fact, there's a picture of the group in Fiji.

Misa: Pretty happy.

Dayan: Anyway, it was a very, very happy project; very successful, did wonders for NZTAB's turnover; their turnover jumped, you know. Everything about it was really very, very good. [Because close of betting was reduced to 15 minutes before race start time and payout allowed within 5 minutes of winners being announced, bettors could now bet race to race with their winnings. This was a far cry from the early days when you bet on all races before the meeting started and collected your winnings the next day. This was one of the main reasons that the betting turnover, and also TAB profit, exploded.]

Misa: So the turnover for New Zealand TAB jumped up then too.

Dayan: Just jumped because, you know . . .

Misa: . . . they're taking the, whatever, 20 percent so that was a gold mine.

Dayan: Absolutely. So then I came back here, which supposedly was as the Manager of Project Management and I had been back here about a month when all of a sudden John O'Neil said, you're taking over all of the wagering projects . . .

Karoly: That's right.

Dayan: . . . because by that stage, they'd sold GWS to Queensland TAB, and they'd also sold it to Natal TAB.

Karoly: Yes, and by the way, it was Queensland TAB; the system where I wanted to get that, you know, that military hardware.

Dayan: Everyone could see, even Jim says it was underpowered. But the point is they kept going with it, there was no end. And all of a sudden, as I said, I found myself here talking to the Victoria TAB, it was Maurie Henderson. So there's your functional spec, meet every item here; and I'm saying, but that's a blue sky document; it wasn't meant to be a functional spec. That's what you contracted for, you do it. That was the attitude of — it was a different group of people at the time — but that was their attitude. Now, I remember saying to John O'Neil, look, I'm very unhappy with this because I just don't think that we can do all of this. By the way, when I first went up to Brisbane, Jim Walters said to me, this is my project. If you try and screw it up I'll get you fired.

Misa: This is in Queensland?

Dayan: Yes. He and Bob Morris were actually based in Queensland. This is the Generalised Wagering System because the whole idea was that the one software system that was going to run . . .

Karoly: Everything.

Dayan: . . . Victoria, Queensland, and Natal.

Karoly: And we thought everywhere else.

Dayan: And we thought everywhere else, yes.

Misa: That was the GWS?

Dayan: That was GWS.

Misa: That was going to be the package, then.

Dayan: The package, yes. Anyway, I was unhappy and I asked John to get someone out to do an audit. So a guy came out from La Jolla [California], and Bob and Jim just completely snowed him. And the answer was no, everything's fine. Heads down, bums up, get on with it. So I was stuck with continuing to do this, and somewhere along the line it became obvious that things were going off the track. Jim and Bob went back to the U.S., and by this stage, La Jolla started to get a bit concerned so they sent out some managers; A.J. Rutter and Ron Hilgers; and there was another guy, I've forgotten his name. I then went up to Brisbane and tried to run that part of it for about close to a year. I was very unhappy, as you could imagine, because they were just adding more and more people, and the old adage, you know, you add people to a late project all you do is make it later. By this stage, we were up to 50 programmers, you know, it was just getting out of control. Meanwhile, NZTAB suddenly says hey, this is going so well, we want a new expanded system; we want to redo all this. So naturally enough, I get involved in it.

Misa: But this is redoing the successful operating system for New Zealand TAB?

Dayan: Yes, they want a new system now, because the initial system only handled the three major areas: Christchurch, Wellington, and Auckland. They said now we want a system to cover the whole of New Zealand, because it's going so well we think we can justify it. Now, comms in New Zealand was very, very expensive because the population is more spread out. I mean, here in Victoria, you know, 80 percent of the population is just in Melbourne. And when RIMFIRE went in, it was just basically Melbourne. In fact, the country areas weren't automated until very much later. But in New Zealand, you couldn't do that; and again, it was the same in Queensland and in Natal; the whole idea was based on just having the major population area coming in.

Karoly: They also had a few very fancy bet types, if I remember rightly.

Dayan: But anyway, so I said to New Zealand TAB: don't go with GWS. And this is when I got approval for these 32-bit computers; this MP32. They were not a standard product [in the "Red Book"] but were classed as a "Special Product." And I actually got them on the build schedule. Now, what I was told at the time was that the MP32 was a skunkworks project out of La Jolla, where these guys in the back room had "dualed up" [sic] the System 17 and made it into a 32-bit computer. And at the time, there was already a 32-bit computer that Government Systems Division was selling to the Air Force. it was a military version of a 32-bit computer. They already had an operating system, and they had compilers; so everything was there!

Karoly: Ended up having to write compilers for the 1700.

Dayan: Yes. So I actually went over to La Jolla; I worked there for about four months; this was in 1977; I was there from about September 1977 through to February of 1978. And I had a guy from New Zealand TAB there, we were working through the functional specification, doing all the bits and pieces to get the project started; and I remember that Chris Reilly came over, and he actually went to Minneapolis and did some work there on making sure that we had the operating system and compilers and all that for this MP32.

Karoly: Yes.

Dayan: And everything was going along well. In the meantime, New Zealand TAB offered me a job, headhunted me and said, we want you to come over and we want you to head up a new division. Not only do we want to automate the whole of New Zealand off-course, we want to automate every single racecourse, as well.

Misa: Wow.

Dayan: And there's 81 of them.

Misa: Wow.

Dayan: And you've got about $50 million to spend. My wife didn't like the States and she, being an only child with elderly parents, she really wanted to get back. So I thought this is all great; I'll get back and I'll do an MBA; I'll sit back and I'll watch Control Data do all this good work; and what have you. And so I left Control Data and came back to New Zealand and this is what started to happen, and I started doing an MBA, and about six months into the project everything fell apart here in Victoria. Victoria TAB cancelled their GWS project; Queensland cancelled their GWS project [laughs] . . .

Karoly: Not unreasonably, mind you.

Dayan: . . . not unreasonably. And Control Data said, do we really want to do this little project for New Zealand TAB? It's going to be an orphan, you know, and we muddied our name in this industry. And they came cap in hand to New Zealand TAB and said we want out of this. How much can we pay you to get out of this? And New Zealand TAB said to me, what am I going to do? And I said well, there are 50 out of work programmers in Australia, I reckon I can pick the “eyes” out of them; and I reckon we can put together our own team. They said go for it, which is what I did; so I didn’t do an MBA. I put together a team, I got all the best guys.

Misa: So you ended up actually working for New Zealand TAB?

Dayan: Yes, I was the Development and Planning Manager for the New Zealand TAB. And what we did was we went out for terminals; we bought terminals from AmTote [or General Instrument as the parent company was called]. And we went out to buy a computer and, at this stage, we said we want a 32-bit minicomputer. We had three proposals, two of which were viable, because Prime, at that stage, was much more commercially oriented, they weren’t really real time oriented. The two computers were the Perkin Elmer computer, and the DEC VAX. And it was very, very close between the two of them, and ultimately we chose the Perkin Elmer 3200 computer. And they had a mixture of them; they had a 3240, which was the large one for the main system; and we chose the smaller 3220 for the race-courses. We could put these in a van and we used to drive them around to the racecourses. The terminals we bought from AmTote were also moved from course to course.

Misa: You couldn’t [inaudible]?

Dayan: Now, the interesting thing — and this is the point that I made — when we bought the 3240, which was a year after Control Data cancelled, it was nowhere near as advanced as the MP32 had been a year before. And my view is that if Control Data had managed to get that MP32 out commercially, they'd have blitzed VAX, because VAX computers subsequently became the *de facto* standard for 32-bit computers.

Misa: They became essentially the *de facto* standard.

Dayan: Control Data had the reputation, it had the hardware, it had the software, and it had the marketing. But why couldn't it get it out? This is where it would be interesting if you could actually go back and interview some people on this, because my theory is there was no product champion, right? And anything in any organization needs to have a product champion. The problem that I could see was this: La Jolla was not interested because a 32-bit [computer] was competition to their 16-bit, okay? Government Systems Division didn't want a commercial version because apparently they had an agreement with government that if there is a commercial version available, they have to drop their price. You can imagine the sort of profits they were making with their militarized version. And that was at the time when Cyber was bringing out the Cyber 71, and they didn't want a low-end, in-house competitor to the Cyber 71. So you had three big divisions, each of which says if this comes out, they're going to hurt my P&L; they don't look at the overall good of the company, and there was no one strong enough within the company to say look, you really need to do this. And I can give three examples that I know of, of similar things that happened to other well known companies. IBM PC. IBM bought their quick and dirty DOS from Bill Gates; they didn't want to own it because they didn't want to compete with their low end machines because they didn't want an in-house competitor. Look at Sony. They had the Walkman, and they had all the technology — this is in Jobs' book — they had all the technology to develop an iPod, but again, they couldn't get it out the door because the Walkman division said this is going to hurt us. And of course, the best example of the lot is Kodak. Kodak had all the patents for digital photography.

Karoly: Did they?

Dayan: Yes, they had all the initial patents. But at one stage in the 1990s, 89 percent of their revenue was coming from film so they weren't about to go to digital. But of course, what happened? They actually went bankrupt; but they've just come out of Chapter 11, just recently. And Jobs actually says in his book, if you don't cannibalize your own product, someone else will. Simple as that.

Misa: And that's of course, a really hard lesson but your idea about their needing product champions, that's just across the board; big companies, small companies. This fell between a number of cracks.

Dayan: Yes, it's a bit sad, really, because, you know, of what might've been. I'm sure that's true of a lot of other things.

Karoly: Actually, given that sort of hardware, GWS could have been a workable proposition.

Dayan: Absolutely.

Karoly: Particularly with some of Chris Reilly's later ideas, which never got implemented.

Dayan: Yes. The interesting thing is that the system at New Zealand TAB is still running today. It's running on different hardware, running on different operating system — but it's still the original software. The same software, it's over 30 years old, same software. It shows that if you have a good design, and a good team, which I really did have, it will last the distance.

Misa: Can you fill in, the New York part, this off track betting. What happened with that?

Dayan: What happened with that, the initial contract was to a company that was tied in with AMP here in Australia; they bought it, basically. Can't think of the name; I forget the name of the company; but anyway, the result was that they couldn't deliver. There has been a lot of disasters in TABs automation, including the Queensland TAB which had two disasters with Burroughs machines; and South Australia had a disaster; I mean;

Western Australia. There's been a lot of systems that didn't work, including the first New York OTB system. And what happened then was that AmTote came in and took over. AmTote used to run all the on-course totes. And at that stage, Control Data had a subsidiary called Ticketron, and Ticketron actually got a slice of the action as well. In fact, Ticketron was the one that had the original terminals that we used the drums out of to make the RIOT terminals.

Misa: That printer.

Dayan: Yes, that printer; Ticketron, yes. So they sort of went their own way and actually had some service contracts to run the OTB in New York. I became quite friendly with AmTote because we bought the terminals off them, for NZTAB and I got to know Jack DeVries pretty well. Subsequently, actually, after I set up my own company I actually represented them here in Australia and my company became their software arm. You know they used to run all the tracks in New York; Saratoga and all of that; on-course.

Misa: New York has a big racecourse industry and then a big betting industry.

Dayan: Yes. In fact, I remember the first time I went there, with AmTote, and coming back on the train from one of the race meetings, with sort of a PR guy. And just at that time, there was a very big scandal going on. You know how they got the illegal betting on numbers; the illegal numbers bets, where you choose three numbers? If you get the correct three numbers, you win $500 I think, is that right? Because you've got one chance in a thousand of getting the three numbers, and basically 50 percent is what you get. And the big scandal was the claim that people who were running the numbers, which was in fact the last three digits of the turnover at the race meeting, were scamming the game. The claim was that they were putting the bets on the winning numbers. I mean, they knew the numbers before the tote was closed.

Misa: Oh, so that's easy, then you just write down that number and you win.

Dayan: Exactly. And the funny part of it was this PR guy…

Misa: But illegal.

Dayan: Yes. But the PR guy that I was with said that Amtote had to write a letter to explain that the reason this happened is that at the course, the computers know the turnover before they publicize it and so the moment that the computer spit out these numbers, these on-course guys would immediately call up their mates down the street [and] say go and put it on 1-2-3. [Laughs.]

Misa: So it's the turnover figure.

Dayan: Yes, it's the last three digits of the turnover figure; that's the number. So it was funny, here are these letters in the paper explaining how an illegal game was being scammed. [Laughs.]

Misa: Well, there was a lot of money in that, I suppose.

Karoly: Diverting you for a minute, do you have anything in your files about the Royal Turf Club of Thailand's system?

Misa: I have two people that have said something about that; that was a similar setup to the TAB system.

Karoly: Factually it was, but in detail, it was very different because they had . . .

Dayan: It was an on-course system.

Karoly: . . . first of all, it was an on-course system, which makes the software technically simpler. But the problem was that it had a lot of electromechanical devices as part of the system. For instance, the system had to operate the results display.

Dayan: That's right, the infield indicator.

Karoly: Infield indicator. And the bet selling machines were electromechanical.

Dayan: ATL machines. ATL was very upset they didn't get that contract, of course, because they had all the on-course systems. And, of course, the reason was that . . .

Karoly: Anyhow, I designed the software for that system.

Misa: Oh, for the Thai system?

Karoly: Yes. And as I say, because of those electromechanical devices, timing was very important. Say it was for things like the display board, it had to be a certain delay between sending the signal, you know, so the whole backbone of the system was a timer – a software alarm clock. Well, programs worked on interrupt basis and a program performing something could tell the time where I want to be interrupted in so many milliseconds. Or, you know, I want to be recalled in 23 milliseconds time because that's the timing of the particular electromechanical device I want to give it another kick or something. So I pinched the CARBINE memory allocation system, you know, these different size memory units being allocated to the different processes as required. But that actually I gave a data flow diagram of the system, now wait a minute, who did I give it to? I think I gave it to John O'Neil last Friday.

Dayan: I think you gave it to me.

Karoly: Did I give it to you?

Dayan: Yes.

Karoly: Did John give it to you or not? I showed it to John O'Neil, the functional diagram; the data flow diagram of the system.

Misa: I got a package of materials from John.

Karoly: That might be in that, because I gave it to him.

Misa: It's in that large manila envelope.

Karoly: This one?

Misa: Yes.

Karoly: Okay, I'll have a look because it was my only copy I have of this huge job.

Misa: John said that these were materials that he didn't want me to take, but that I needed to copy. But I don't remember seeing a diagram.

Karoly: No, it's not there. I wonder what he did with it.

Dayan: Or did you give it to Ron? Because Ron was the project manager of that.

Karoly: Maybe I gave it to Ron Bird. I couldn't remember whether I gave it to John or Ron, maybe it was Ron. Have you seen Ron yet? Ron Bird.

Misa: Yes, I talked with him on Tuesday.

Dayan: My only involvement with the Thai project was one stage when I was Manager of Project Management. I saw all the financials and I remember being very naïve, saying "Cost of Sale - 20 percent! How can that be." I was told shshsh, that's the "General's cut." Okay.
[Laughter.]

Karoly: Oh, yes.

Misa: Are there any other topics that we could usefully record this afternoon.

Dayan: Not really.

Karoly: No, I don't.

Misa: This has been very rich, thank you so much for your time.

Dayan: This is the write-up on those three projects. It's actually on the web.

Misa: Is that on the [CDA] website, as well?

Dayan: I think it's on the website because the photos are probably better.

Karoly: I left Control Data later because the Systems Division died, and they didn't seem to have anything for me to do.

Misa: Yes. Then Marcel, you moved on to the New Zealand TAB?

Dayan: New Zealand TAB. I spent the first five years putting in this on-course/off-course/ telephone betting system covered the whole island and worked myself out of a job. And basically found that I was getting bored; I was picking fights; I said it was time to leave so I came back here and starting a consulting company. We then moved into developing Keno systems. AmTote became General Instrument so we became the software arm for General Instrument. We developed Keno for the Jupiter's Casino; and then Keno for Macau; and then we did Keno for the whole of New South Wales [NSW Club Keno]. And then we found that we were making other people rich and we thought it was time to do something for ourselves, so we developed our virtual horse-racing.

Misa: So it's a software package . . .

Dayan: It's a software package, yes.

Misa: . . . that included the hardware for free.

Dayan: Exactly. And unfortunately, the TAB here started to privatize just at the time we were supposed to go with it, and so we were stuck with a product and no market. So I then went round the world selling it to all sorts of Third World countries, and have some very interesting stories to tell. And then in 1994 we went out and got some venture capital; and the actual venture capital was from the Australian government. And that kept us going but we didn't go viral; and we didn't die; we just plodded along.

Misa: Middle ground.

Dayan: Middle ground. So after five years, our VC partners said it was time to get rid of you and there's the TAB sitting there, just waiting. Of course, they then scooped us up and we didn't get very much for our product, as you can imagine, because they were sitting there waiting, as they had the license and could dictate terms. I had to do an earn out for three years, that actually worked out pretty well. They are making so much money out of the product now; they're probably making about $100 million, bottom line, a year. At one stage we almost got it into the UK but we didn't want to sell it outright, so Ladbrokes went behind us, took all our materials and ideas and got someone else to develop it. Their turnover about four years ago, was £2 billion a year. And this is a product when everyone told us we were stupid to develop it, saying nah, who's going to bet on virtual horse racing?

Dayan: If you go to any TAB you'll see it running; it's called Trackside.

Misa: Trackside, okay, I'll have to check that. George, you said that you left Control Data. What did you do after that?

Karoly: I joined Computer Power. None of it still exists; it was a software consulting, fairly small outfit. Actually, it was a mistake on both sides

Dayan: They was trying to bulk it up so they could float it.

Karoly: Yes. Anyhow, he wanted me to set up a consulting division in this company, which up to that stage, was mainly doing software development. And I thought he wanted me on the technical side but he thought that he was getting a salesman, which I'm not. So I left after a couple years and went to the Victorian Railways. Actually, what was my title? Manager of Technical Support of their computer center.

Dayan: He floated the company and made a lot of money, and then died. [Laughs.]

Karoly: He claimed that if he ever made a million, he'd retire and write a book. And that's the last; well he made more than a million, but he never wrote the book. [Laughs.] Maybe it's still coming, I don't know.