A. M. Turing Award Oral History Interview with
Niklaus Wirth
by Elena Trichina
ETH, Zürich, Switzerland
March 13, 2018

Trichina: My name is Elena Trichina. It is my honor to interview Professor Niklaus Wirth for the Turing Award Winners Project of the Association for Computing Machinery, the ACM.

The Turing Award, sometimes referred to as "the Nobel Prize of computing," is given annually for major contributions of lasting importance in computing. Professor Wirth received his award in 1984 in recognition of his outstanding work in developing a sequence of programming languages – Euler, Algol-W, Pascal, Modula. The hallmarks of Wirth's languages are their simplicity, economy of design, and high-level engineering. We will talk about it and many other things later today in the interview that takes place on the 13th of March, 2018, in the Swiss Federal Institute of Technology.

So we start. Ready? Guten Tag, Niklaus. Gruetzi.

Wirth: Добрый ден, Елена Василевна Здравствуи *[Good morning, Elena. Hello. – ed.]*

Trichina: Well, although your command of Russian is sufficient, my command of Swiss German is a little bit rudimentary. Shall we continue in one of the languages where we both feel comfortable? What would it be? French? English?

Wirth: Oh, I think English is better for me. Yes, yes.

Trichina: Okay. It's probably better for the audience. [laughs]

Wirth: Okay. True.

Trichina: Okay. We continue in English. Here in front of me there is a list of the most important days in your life. You were born in February 1934 in Winterthur, which is a sort of middle-size industrial town in Switzerland. Can you tell a few words about your upbringing, your parents?

Wirth: Yes, of course. I should say I was born in 1934. My parents were at that time already over 40. My father was a geography and geology teacher at the local *Gymnasium*. My mother took care of the household and the cooking. And I

was left to my devices. I was left undisturbed and pampered. Yes. And I went to school, primary school.

Trichina: And now my question about the primary school. The years of your primary school coincided with the Second World War. Do you have any recollections of this time? Of course Switzerland was neutral, but still.

Wirth: Yes, yes. Of course. I remember that my father, who was a soldier in the army, was often at the border guard looking over the Rhine and seeing the German troops amassing. So, very often he was away. It was a difficult time. In spite of all, we had enough food still, but it was unpleasant. Pleasant was, in the later part of the war, that there were always airplanes going above in the sky, and I became a really small airplane enthusiast. I knew all of them by sight. And one morning I even saw a squadron of Liberators flying over the city, and half an hour later I heard on the news that had bombarded the neighboring town Schaffhausen.

Trichina: Oh.

Wirth: Yes. Zurich was also bombarded once. The railway station. It never became clear whether it was accidental or on purpose.

Trichina: Okay. I suppose the formative years were the years when you were at the *Gymnasium*.

Wirth: Yes.

Trichina: Did you have any favorite subjects?

Wirth: Well, since my really early childhood, I was fascinated by technical matters. There were four railway lines going past my house and I always watched them and became an expert, setting up timetables and so on. So railways. I wanted to become a locomotive engineer. But then after a while I switched to wanting to become pilot. That's a very early youth experience. Yes. And then of course I started building model airplanes. That was so to say my career. I was fully engaged in this. I have no idea how I mastered going through *Gymnasium* so to say at the side.

Trichina: [laughs]

Wirth: But my major subjects were certainly mathematics and physics and chemistry. I didn't do too badly at languages. We started with Latin as the only foreign language in the first class of *Gymnasium*. In the second year, we started French. In the third, English. In that sequence.

Trichina: If your hobby and love was airplanes, how come that you decided to study electrical engineering?

Wirth: Yeah, that's a good question. Starting in the *Gymna-*… Well, in the middle of the *Gymnasium*, I became a real enthusiast about model airplane building. I even was elected to go to international competitions in England. Then around 1950, radio control equipment became available, and I think I was one of the very first in Switzerland who ordered one from England. The thing of course never worked. They were very unstable. Nowadays of course I know why. So the only thing was to dive into this subject and to learn how these things operate and how perhaps to fix them. Then the years where I had to choose a study approach, and I decided on electronics because…

Trichina: So it was not electrical engineering? It was electronic engineering?

Wirth: Well, in my mind it was electronics. But at the ETH, they were one thing. And, as a matter of fact, electronics and power engineering separated only in the fifth semester, and in the first four semesters it was mostly mathematics and physics and machine designing. And I think we wasted a hell of a lot of time on these subjects. Not mathematics. That was the thing that I always appreciate and it was of lasting value. But the first lecture I attended in electronics was in the sixth semester.

Trichina: Third year?

Wirth: The third year. And that was too late.

Trichina: Apparently it wasn't. It worked well for you later in your career, this…

Wirth: Yeah, yeah. But to tell you the truth, all that I learned that was of later use in the subject I had learned at home from books, doing things, experimenting.

Trichina: Of course. But the foundation is always necessary.

Wirth: The foundations were rather in physics.

Trichina: Mathematics.

Wirth: And mathematics, yes.

Trichina: But also probably design lessons were not such a waste of time, because you never know. Life moves in mysterious ways.

Wirth: Yeah. But all these practical things I learned at home, from friends.

Trichina: Okay. The difficult questions. After having finished the Swiss University of Technology, you decided to emigrate. To Canada was it?

Wirth: Yes.

Trichina: What prompted this decision?

Wirth: Well, first of all, I should praise my father who had suggested that I go abroad right after studies for a year or two to see something of the world, much to the sorrow of my mother of course. And I picked this up particularly because my best friend who was three years older also emigrated to America a year ahead. And I got married *[0:10:00]* and four days after marriage, we went to Paris and to Le Havre and boarded a boat, which took seven days to New York, and my friend fetched us there. So that was…

Trichina: Your friend, Mr. Brack. *[Walter Brack – ed.]*

Wirth: Mr. Brack, yeah. That was a real experience. And today I am amazed at my courage of that time. It was a complete break of life, being married…

Trichina: But you had your wife to cook your breakfast. [laughs]

Wirth: Right. Yes, yes, yes.

Trichina: Yeah, I agree. Of course it was an extraordinary step at the time.

Wirth: I had to leave all my workshops, all the electronic gadgets, everything.

Trichina: [laughs] Okay. So it was Canada. What did you do there?

Wirth: Well, I actually wanted to go and pick up a job as an engineer, but that was difficult at the time from the distance. But somebody told me I could get an assistantship at the university in Quebec, Université Laval. And I said to myself, "Well, I accept it. The salary is absolutely lousy. But if I don't like it, I'm at the place and I can find something else." They had just started an electrical engineering graduate department and we were no more than five students. One Chinese lady, one Brazilian major, one Frenchman, and one… I think one… I forgot where, and me. Yeah.

Trichina: [laughs] So it was where you were exposed to computers for the first time?

Wirth: Well, you see, maybe I should say I had taken a job in Switzerland as an engineer, and it was an absolute disaster. I realized that I had taken the wrong job. I couldn't sit all day from morning at eight till five in the evening in front of a workbench and improving a certain amplifier a little bit. And I quit after a month.

Terrible. But I then had this proposition of going to Canada, and I thought it was nice. I realize that actually my calling was that of teachers, like my father.

Trichina: So early in your life you already…?

Wirth: Well, that's not so early. I was 22 or 25. Anyway, I thought I had to do to go into teaching, and that was in Switzerland not possible with an engineering degree. Everything was compartmentalized. And so I thought, "If I go to university, I should take some courses in math and physics," and so I did. I took one course in numerical computation from a professor by the name of Goodspeed.

Trichina: [laughs] A fitting name.

Wirth: And they had one of the first experimental computers with tubes, and its main characteristics, whenever we had finished one exercise, it had broken down. It was always in the hands of the technicians. And, for the whole semester, I couldn't run a single program. Well, this was the state of the art at that time.

Trichina: Yeah. You were lucky just to see a computer anyways at the time.

Wirth: Right. Yeah, yeah.

Trichina: But you didn't stay in Canada for very long.

Wirth: No.

Trichina: So what did you do next?

Wirth: I stayed exactly 12 months, and then we headed west with a short stop of four months in Ottawa at the National Research Council where I built a small analogue computer for computing autocorrelation. Then I took the plane. We had just saved enough money to pay a ticket to California. It was the first time…

Trichina: Just like that? Just flying to California with no…?

Wirth: Yeah. Well, no, no, no, no, no. I had been encouraged by my co-students to go on for a PhD. They said, "Well, just a master's is just not good enough." After a long hesitation, I finally thought they were right and I applied at four or five universities, and picked the positive reply from Berkeley.

Trichina: These three years at Berkeley, what they were like?

Wirth: They were fantastic. It was really a revelation. I had been used to the environment, the drab environment of ETH in Zurich with courses that were uninteresting, with teachers you never met, with colleagues whom you hardly met

because I was rushing home by train to go to my workshop. And of course in Berkeley… well, also after this really stiff winter in Quebec, Berkeley was fantastic. The climate, everything was right. Blue sky, sunny sights.

But a surprise was that I was asked to take more courses, more classes in math and statistics. My education from ETH was not considered sufficient in this respect. So I did take classes where we had exercises every week that were graded and I got good marks. So the dean secretary once asked me, "Have you ever considered asking for a scholarship or a fellowship?" "No, no." It never would have entered my head. And she said, "Well, here you have the forms. Apply for them," and I got a Ford Foundation Fellowship. That speeded up my progress towards a thesis quite considerably.

Trichina: How did you choose the topic of your thesis?

Wirth: Ah. [laughs] Well, I did a lot of wandering around. I first had an assistantship with a professor in information theory. After a few months, I found that this was really not my cup of tea. It was far too mathematical and nothing else. Then an opening became open under a professor who was dealing with computers. One of the early pioneers, Harry Huskey. He had for instance designed the Bendix G-15, and that became available. Was absolutely terrible to operate, but in these early years, you just took what there was. And nowadays I'm rather surprised that people could ever get enthusiastic about these poor gadgets.

Anyway, I worked with this Bendix G-15, and at the side I learned Fortran on the Computer Center's IBM machine. But of course I had to look for a topic for the thesis. There was a guest professor from Amsterdam coming in, and I associated pretty much with him because he had really original ideas, *en vogue* for us. I found a small group of graduate students working in an office in the cellar, and they wrote a program which translated a dialect of Algol. A dialect of Algol 58 to be precise. And that program was written in the language that they translated. Oh, very fascinated.

Trichina: And indeed, even now it sounds fascinating.

Wirth: Yeah, yeah, yeah. It was the language NELIAC, an Algol dialect, Algol 58 dialect. And it was really chaos. It was hardly intelligible. It was one woman who seemed to understand a large part of it.

And I dug into it. I wrote my first little Algol compiler, Algol 60 compiler, and I thought, "Anyway, here is a subject in a big mess. It would be a worthwhile project for a PhD thesis, which has to have some originality supposedly. It would be appropriate to pick up the task of getting some order into this mess." That is of course how I got into programming languages and, totally parallel, of compiler

design. *[0:20:00]* Because a programming language without compilers is worth hardly anything, as Algol 60 had proved.

Trichina: [laughs] So the problem of automatically translating the programs in high-level language into machine code intrigued you?

Wirth: Yes.

Trichina: Intrigued or you thought it was a practical program to address?

Wirth: Yeah, yeah.

Trichina: Both.

Wirth: Yeah. I had of course used this Bendix machine, and I even on the IBM did some assembly coding, and I immediately realized that was not the way of the future, that it's so error-prone and time-consuming that a better tool must be used. And these were obviously the programming languages as shown by Fortran and Algol.

Trichina: I want to give a quote. In his 2010 interview, Edsger Dijkstra said, "Computing science started with Algol 60. Now the reason that Algol 60 was such a miracle was that it was not a university project but a project created by an international committee." Would you agree with this statement?

Wirth: Yes, but the members of these committees, not all but most of them came from universities.

Trichina: [laughs] Okay. So you finish your PhD, and almost immediately you became one of the members of such a committee, didn't you?

Wirth: Yes. I happened to get an assistant professorship at Stanford University. Actually we were only five professors, two of us assistant professors. And I was fortunate I could simply continue work on my thesis. We had a fantastic computer, a Burroughs 5000, there. Famous, but soon extinct. And so I could continue this work, and it made me known in this committee where this professor visiting in Berkeley was also a member, a prominent member. So I was invited to attend one of their meetings, and it became a frequent transatlantic hop, which I found terrible. But anyway, I participated in these meetings, and that's how my work flowed into this group.

Trichina: People with whom you worked in this group, such as Tony Hoare, Edsger Dijkstra, Peter Naur, have you been professionally or personally in touch with them through your life, through your working career?

Wirth: With Tony Hoare and Edsger Dijkstra, yes. I also visited them quite frequently. Once I spent three weeks in Belfast with Tony and Jill, hearing the bombs go down in the city. With Dijkstra, I visited a few times in Texas. And I spent two months in Amsterdam, although he had gone there already. But Professor *[Adrian]* van Wijngaarden was the director of the center. And I had received the duty, the charge by this committee to draft a proposal for a successor to Algol 60. I went back to Stanford, worked on that, and then I think in October I came back to the meeting point and presented my proposal.

Trichina: And now we are approaching to this famous schism in the Working Group 2.1.

Wirth: Yes, exactly.

Trichina: Are you willing to talk about it?

Wirth: Of course, of course. We are not here to hide secrets. [laughs]

Trichina: Right. [laughs]

Wirth: Yes. As I say, I worked two months in Amsterdam and had frequent meetings with van Wijngaarden. And two or three months later we met near Grenoble and I presented my proposal. And, to my utmost surprise, he stood up and presented his proposal, of which I had had no idea.

Now to make the story brief, the difference between those proposals reflected itself in the aptitude of certain members of the group. There was one group who proposed as it was then called Algol X, something that should be practical, have only more or less proven facilities. It was urgent to get a replacement of Algol 60 – there was PL/I, the big competition, and so on – so we couldn't wait for years. And my proposal was exactly this line. Hoare and Dijkstra were also of this opinion. And the other faction, called Algol Y, wanted to set like Algol 60 another milestone. And the critics afterwards called it a "millstone."

Trichina: [laughs]

Wirth: Yes. And it had to be another big thing like Algol 60 with fantastic facilities, flexibility, and so on. And it became very, very difficult to implement. These ideas had never been tried before, and it didn't become an Algol 68, the first compiler became available something like '72.

Trichina: And this is how Algol-W came to existence?

Wirth: Yes. I want back quietly to Stanford – that was in 1966 – and built up a group of graduate students, and we implemented my Algol X, and it was later called Algol-W. Implemented on IBM machines and it became quite widely used.

Trichina: Could you say a few words about this mysterious language PL360?

Wirth: Oh. Yes, yes, yes. Of course. That was… Well, I had to ask myself, "With this Algol-W, how is it going to be implemented? In what language?" At ETH, we had only an atrocious version of an Algol compiler, Fortran, assembler code, and they both were out of the question. So I designed this language PL360, which was very close to the machine code but had some structure. And I used that to write the first compiler, the core of the first compiler and then bootstrap it up from this. So from the beginning it was only intended as a provisional tool for my own purposes, however there were quite a few interested parties around the world afterwards and it became used here and there. But I forgot about it very quickly.

Trichina: Okay. I think, was it 1967 when you created Working Group 2.1 and never joined again?

Wirth: Yes.

Trichina: This brings us to '67-68 when you left United States and returned to Switzerland. What prompted this move?

Wirth: Well, I received an invitation to become an assistant professor at the University of Zurich.

Trichina: At your alma mater?

Wirth: No. My alma mater was ETH. So it was the neighboring, across the street, the neighboring institution.

Trichina: Ah yeah, university.

Wirth: And I thought, "Well, why not? Why not try? I have never been working in Switzerland except that one month. And we could visit the parents and see how it is." And frankly I was on the verge of packing the suitcases again, going back, because I had just asked for a year's unpaid leave at Stanford. But then this offer came from ETH as an associate professor coupled with the promise *[0:30:00]* that now a computer science department would be founded. That was very enticing because I always had had in mind to go abroad to learn something that cannot be learned in Switzerland itself and bring it back.

Trichina: And bring it back.

Wirth: Well, that was so good, so fine. But to tell you the naked truth, it took fully 12 years until ETH finally established a computer science curriculum. Twelve years!

Trichina: Okay. But in the mean time during these 12 years, what did you do at ETH? What did you teach?

Wirth: Well, we taught programming courses and elementary computing courses at other departments like electrical engineering, like physics, like most engineering departments. So that kept us busy. And of course, I had time to do research, so that was positive.

Trichina: Pascal had become so widespread, and I think many universities from Switzerland to Germany to Russia to China to India were teaching Pascal, or rather teaching computer science using Pascal as a pedagogical tool. For example, here I have a Russian book ПАСКАЛЬ - Программирование на языке высокого уровня, so "Pascal - High-Level Language Programming". And even now, it was yesterday when we were in the computer science department buildings, building in front of the Lilith, it was an Indian student who came along and couldn't believe that he can see the creator of Pascal because he studied in India computer science via Pascal. So it was a huge I think success.

But there was one more intriguing property of Pascal, at least intriguing for me, for the Pascal project I mean. To help implement Pascal on computers of all different kinds, not only on one kind, it seemed that you had taken the approach that now is widely known as "virtual machines." Right? You wrote a compiler from Pascal to a computer, a hypothetical computer, and then to port the code from this computer to an actual one, whichever it would be, was an easy thing. Do you think it was the first time when virtual machine came to existence, the idea of virtual machine?

Wirth: It was a technique known, but not widely known. But for instance my earlier Euler language was also implemented in exactly this way – that you postulate the hypothetical instruction set and then implement that as an interpreter. That's usually a small project. You know, it can be done in a few days. And this is indeed how we helped other universities adopt Pascal. They asked us and we said, "Well, we can send you our compiler, our P compiler – for 'portable' – and you simply have to write for your machine an interpreter for that p-code." And that helped enormously spread the language.

Also, I must say Pascal was released report in 1970, and in '72 for the first time I used it in teaching to large, 300-people classes. But the actual breakthrough came several years later, namely around '75, when in America the microcomputers had started spreading and intruding into schools and households. They of course had to have a compiler that was reasonably small because they had small memories, and Pascal compiler with this p-code fit into the small memory. That's where the company Borland in California picked up…

Trichina: Turbo Pascal.

Wirth: …Turbo Pascal, exactly. And the big thing was of course they sold a diskette with the whole system for $50.

Trichina: [laughs] No, really?

Wirth: Yeah, yeah. Whereas compilers on mainframes cost $10,000 and more.

Trichina: Okay. During this time, during '70s, you were also engaged in work on programming methodology, more or less formal methods of program development. Can you say a few words about the method of stepwise refinement?

Wirth: Not much. It was of course… for me, it was an natural method. You start…

Trichina: Can you explain it?

Wirth: You have a certain task, and because there's not a single instruction on your computer to solve it, you break it up into subtasks. And this you continue until you're down at machine instruction, or programming language statements, rather. But of course you need certain principles and discipline to do that, and I demonstrated this on I think two examples and published it in 1971. That became a fairly well-known paper on the subject.

But of course the main flow already at that time was towards more formalism – Hoare triples, Dijkstra predicate transformers. And I used these things in my classes too, but I never believed that they would be viable when you design huge systems, which from the beginning are not that precisely specified.

Trichina: Anyways, so now we are at 1976, which was also a very important year in your career because of your sabbatical.

Wirth: Yes.

Trichina: Can you elaborate?

Wirth: A sabbatical was coming up, and then the question is where do you go? And I felt just going to another university and doing the same teaching as you did at home is not the right solution. Why should I go somewhere else to do the same thing? And a year earlier I met Butler Lampson, whom I still knew from Berkeley, at the conference in Los Angeles, and approached him and said, "Hello. I've heard you are the Xerox lab and doing very nice things, very interesting things. Do you have a program for sabbaticals?" "I'm afraid we don't, but I could look into this." And a few months later, I got an invitation and gladly picked it up. So in the summer, August of '67 *[presumably 1977, ed.],* I bought a Volkswagen camper bus, imported it to Switzerland, packed all the suitcases, packed my wife and the three children all in that little car, and we headed

towards Antwerpen it was, and then across the Channel, England, and in Southampton we boarded the Queen Elizabeth 2 boat, which took us in seven days to New York.

Trichina: So it was your second crossing?

Wirth: Second crossing on the sea. That's right.

Trichina: And the outcomes of this sabbatical? Impressions, projects, outcomes?

Wirth: Yeah. There were so many groups and excellent people at that Xerox lab that it took me a long time to decide what I really wanted to concentrate on when I returned. But after about six or seven months, it became clear to me that I had to have such a personal computer like they have developed with the Alto. I couldn't possibly go back to Switzerland and work through a very thin line with that huge monster in the cellar. No interaction practically. And, well, what do you do? I mean these Altos weren't on sale. They couldn't be bought anywhere. Xerox had developed them for their research lab, period. And like with the compiler, *[0:40:00]* what do you do in that case? Build it yourself. If you're an engineer, you can do it.

I was then fortunate to get in touch with an assistant professor of Utah who was teaching hardware development, relatively simple things. He called me and wanted to know whether it would be possible to come to Switzerland for a PhD in software. He would like to know, to learn about software. And I thought for a while and said, "Well, and we want to learn about hardware. Might be a good exchange." He actually came with his family for three years, and the result was the Lilith computer modelled after the Alto with already a bit more modern technology.

Trichina: But the Lilith computer was mostly for teaching purposes, right?

Wirth: Not…

Trichina: Or you had more ambitious plans?

Wirth: Yeah, it was… Yeah, we built 20 of them, and 10 of them were put into a student lab. But the 10 others were in our institute for the researchers. Not only in my group, also in Zehnder's group. And it was not just a PC toy. It was the first time I saw such a personal computer with which you could solve real problems, including building a compiler for it. They had large enough memories, enough speed, you could really work.

Trichina: Yeah, but your approach to the Lilith project was somewhat in the opposite direction at that time, because you had chosen, okay, one user, it's

okay, but one language, one operating system, one compiler, which means that it was… again I understand that it was for the purpose of doability, right?

Wirth: Of doability, exactly. I mean I had Richard Ohran, myself, and two or three graduate students, and we were up to building a whole computer and software for it, whereas companies employed armies to do such a thing. And the only solution was to confine yourself to modest goals, such as one language, one user.

Trichina: And what language was it? It was not Pascal.

Wirth: That was Modula.

Trichina: Multithread?

Wirth: And Modula is kind of a successor to Pascal, more extended for system design, system programming. Maybe I should say again, at my sabbatical at Xerox, I got to know the language Mesa. That's their company internal research tool. Again, not available outside. And it was very much built on top of Pascal, but bigger. So my obvious solution was to take Mesa as an example and reduce it again, and this is how Modula came up, with the idea of modules with separate compilation.

Trichina: So, was it a revolutionary idea of yours? Or you just picked up on some other developments and just made it a cornerstone of your language?

Wirth: Well, you see, a language designer picks ideas from others, facilities, features, and has the task of putting them into a coherent whole, into a framework. And that is less easy than people might think, because you want to have as much power in the language as possible, yet a compact structure and compact syntax. And that you can get I think only through experience with language design.

Trichina: Is language design also, with system design, was designing compilers for the languages?

Wirth: Yeah. Right. Exactly. That had always been on top of my mind, and I had learned that in the early Berkeley days. I had learned how not to do it.

Trichina: Through?

Wirth: Well, through that dialect compiler of NELIAC. Yeah. Often you learn a lot through negative experiences.

Trichina: 1980 was a year of very impactful events in your life. On the one hand, the elation of getting first homemade Lilith machines. On the other hand, personal upheavals. Do you want to talk about this?

Wirth: Yes. Why not? I separated from my wife in early 1980. The difficult year was really '79. But I moved out, I went to a one-room apartment for a year, and I had gotten to know another lady with whom I moved together a year later, two years later. So that was indeed quite a stressful time. Very bad, really. A traumatic experience. And I must say, Lilith and work and teaching had kind of upheld me. That's where I could forget all the other troubles.

Trichina: But you kept in touch with the children? You had three by the time.

Wirth: Yes, yes. It was not a very hostile separation or divorce. I think we were both reasonable people, and if you see that something doesn't harmonize anymore, there's not much to do. Of course you can make a lot of compromises, but we didn't. And we separated in decent agreement. And the children, yes, particularly now in the last 10 years, I have excellent contact with them.

Trichina: Okay. On a more cheerful note, the first 10 years of your work in Federal Institute of Technology culminated in the creation of the Department of Computer Science, finally. Was it a one-man effort? Or did you have collaborators who contributed a lot to it?

Wirth: Yes, of course. It wouldn't have been possible as a one-man show. I'm not born as a big fighter, anyway. The one man who played a major role was Carl August "Gusti" Zehnder. He joined about two years after I joined ETH. And we had made a proposal for a new department as early as I think about '71. And it was rejected. And another one in '75, which was also fruitless. It was not so much the school board or the presidency was against. Actually, they were quite understanding and favorable. No, it was mainly industry, which said, "Well, we don't need computer science. Programming can be learned by any decent engineer at the side in a few days. So we need mechanical-electro engineers, but not computer scientists." And that's how it ended. Also, the students didn't show interest. But anyway.

Then towards 1980, ETH got a letter from Bern, which is the federal government, saying, "Well, we have several delegations coming and complaining about the lack of programmers in our country. What are you planning to do? What about ETH? Have they been sleeping all the time?" Yes, that was kind of exhilarating.

Trichina: And how many students did you get first year?

Wirth: The first year, we started with a course in the third year, *[0:50:00]* not at the beginning. And, if I remember correctly, there were about 70 students. Quite a lot.

Trichina: Oh. This is considerable.

Wirth: And in the second year, there were 150, and then third year, 200, something like that. Of course, you can't just not bang on the table with a hammer and there it is. You have to design a curriculum, you have to acquire teachers which are capable, and that was the most difficult thing. Teachers trained in computer science simply didn't exist. So in the beginning, we were essentially… well, we were three professors. Two of them had been at ETH already being mathematicians. So I think in all honesty I can say I was the only computer scientist there. And then one of them died after two years, that was another complication. Then Zehnder came in, made these proposals.

Trichina: Okay. Around this time, 1982, you made your first trip to China.

Wirth: Exactly. Precisely. Yes, quite so. They asked me, the school board asked me whether I would be willing to participate on a delegation of ETH professors to China, because they had received several delegations from China here and the Chinese wanted to not retaliate but give something back. And we were six professors. And it was kind of a strange experience. We had of course every… almost every evening, there was a banquet somewhere – Beijing, Shanghai, Wuxi, Nanjing, Guangzhou. And in the middle of the trip, we started feeling a bit uneasy because we were only consuming banquets and giving nothing back. So we asked, "Listen, what's the matter? We could for instance give a lecture, or two lectures." They said, "Oh, yeah, that's wonderful idea. Oh, great." And next day, there were two lectures with at least a thousand students. They just popped out of the ground, you know? But it was a dull experience because we didn't get any feedback, no questions.

Trichina: It's very hard to get questions from a 1,000-strong audience. [laughs]

Wirth: The real difficulty was that they didn't understand English.

Trichina: There was no translation?

Wirth: No. It was 1982. China was still a closed country.

Trichina: And not so many people would speak English?

Wirth: Sometimes we were treated like foreign… like extraneous guys from Mars or God knows where.

Trichina: [laughs] Well, maybe not exactly like this.

Wirth: It was very interesting and we were very, very well treated. Very well treated.

Trichina: Okay. And now 1984, the Turing Award. What was your reaction when first time you have learned that you are a winner?

Wirth: Yeah. I was of course very pleased. But I didn't jump over fences and things like that. It happened to be that I was on a second sabbatical at Xerox 1984-85, and the prize was given out in San Francisco, so it was just a short hop for me. But it was a very big event for me. Of course in Switzerland, that prize is not known. Or was not known, for sure – things have changed a little bit now. No, I was very pleased. But there is not the same thing as in America where immediately you get famous and have salary raises and all that. There was nothing of that sort at all.

Trichina: [laughs] Who was the first who congratulated you?

Wirth: Oh, I guess my wife.

Trichina: [laughs] Diana.

Wirth: Diana, yeah. My second wife. Exactly.

Trichina: Okay. Was it difficult to prepare a speech for the Turing Award ceremony?

Wirth: It was… Yeah, it was not terribly difficult, but it was work. And at that time, the Turing Award lectures were articles that were published in the *Communications*. I think that's not always the case anymore. I'm not sure. No, it was a big affair and it was a big audience of course. You had to be careful doing a good job.

Trichina: Okay. Second sabbatical and its impact on your future work, on your work after this in…?

Wirth: The influence was not that dramatic as the first one, of course. A new generation of computers had come in in '75, and today I call the years around 1975 "the beginning of the computer era," nothing like Babbage and Leibniz and so on. They were not much to do with what we do with computing. But in '75, these powerful workstations were created which are in use on every desk everywhere in the world nowadays, and people don't have an idea anymore how computing was done before – through thin lines going into a supercomputer somewhere or even submitting card decks.

Trichina: [laughs] Well, it was a long, long time ago, huh? Okay. You came back and started your Oberon project.

Wirth: Right. That was, on the scientific side, the output of that second sabbatical. Yes. And, you see, we had been using Modula, had built compilers for it for PDP-1's and smaller computers. And now… Then I had… Well, and then again, like in the first sabbatical where I got in touch with Mesa, this time it was called Cedar, another language, very complex and powerful, for their newer super-efficient computers like the Dorado. And again, my stringent feeling was "That thing may be fine, it was certainly developed by competent people, but it's just too complicated. Unnecessarily complicated." And I said, "You have to do the thing as with Mesa. You have to create a language which just concentrates on the essentials and leaves the rest out." That's what I did, and the result was Oberon, a very dense language with a precise report on 16 pages.

Trichina: But the book on Oberon… Oh, it is the book on Oberon System which is pretty thick.

Wirth: There is a book on the Oberon System. That is of course about editors and file systems and all kinds of things, written by *[Martin]* Reiser. I think you refer of that. But the text programming in Oberon I think is less than ten pages, a tutorial.

Trichina: So what are the hallmarks of Oberon language and Oberon System? How you can characterize it? How you can describe it to the audience?

Wirth: Well, I think it is a nicely coherent system with simple rules yet powerful features. Of course it could never have been our goal to compete with commercial systems like Windows and C# and things like that. We didn't want to. As far as power is concerned, I think we are good enough.

Trichina: So Oberon language and system was also created mostly for teaching purposes? Or did you envision some wider audience?

Wirth: Well, by that time, my ambitions had become more modest. *[1:00:00]* There was obviously no way to play an influence next to Java and C# and C++, if you have to mention it, an awful thing. But yes, I thought, as I did in the Pascal times, the foundations have been laid when a person learns his first programming language. That's the key. In order to do a good experience, you have to have a clean language with a good structure concentrating on the essential concepts and not being snowed in. That is my primary objection to the commercial languages – they're simply too huge, and they're so huge that nobody can understand them in their entirety. And it's not necessary that they're so huge. But people don't want to see that. Particularly industry doesn't. I think the industry thinks it's a big asset to have a complex thing that increases their reputation of being sophisticated.

Trichina: Maybe not so much complexity as being all things for all needs and all people? [laughs]

Wirth: Well, yeah, and complexity's the consequence.

Trichina: And complexity is a consequence of it. Yeah, well… And of course the Oberon workstation, it had graphical user interface, it had all the utilities, connecting with printers and so forth. So, who did all the work? Who did all the programming? Who wrote all this system software?

Wirth: You're alluding now to the workstation series?

Trichina: Yeah. I wanted to lead from Oberon to Ceres.

Wirth: Yeah. Ceres was really built for Modula and for Oberon. I felt we should retain the ability of teaching and working with hardware in our computer science department. I thought that was an important aspect. So we decided, or I decided to build another personal computer, but this time with a commercial central processor. And it succeeded quite a bit. There were actually three versions produced by our technicians. Ceres-1 with a 10-megahertz processor, Ceres-2 with a 25-megahertz – you see, we were going with the times – and then Ceres-3 was probably the most interesting, a computer, single-board, completely without moving parts. So we had 60 stations in the student lab and they work all together, no noise. I think it was the first station with that quality.

Trichina: Who manufactured those stations?

Wirth: Well, the boards were manufactured at an electronics company in Winterthur. A small company. And they built about a hundred of them.

Trichina: And again, Ceres is under exhibition in the computer science department building, and we have a photo of Ceres, how it looked like, and I hope the audience also would like to see it.

Wirth: Yes. The chief architect was Hans Eberle on that project.

Trichina: Okay. Who is now with NVIDIA and for many years he was with Sun Microsystems in California.

Wirth: That's right. He was with DEC for some time.

Trichina: Ah. Initially, yeah. Okay. Now comes your first trip to Russia, 1990, the time of *glasnost* and *perestroika*. Do you want to say something about this trip to unknown territory?

Wirth: Oh yeah. I'm burning to say something about it. It was… I retired 1999, so that was shortly after my retirement.

Trichina: No, it was 1990. Bukhara, Samarkand, Novosibirsk Akademgorodok.

Wirth: Ah. I'm sorry, Elena. Yes, yes, yes. The big trip came later. Yeah, yeah. 1990, I got invited by Andrei Ershov in Novosibirsk, the one who was the exponent of Russian computer science in the Cold War years. And although I had always felt little inclination to visit Russia with that awful Kremlin and… but then it was Gorbachev time, so some loosening up, and I thought it would be the time now.

And I went to Moscow. Just the airport, changing from… What's the name again? Shemya…? Anyway, going through the whole city to Domodedovo…

Trichina: Ah. From Sheremetyevo to Domodedovo.

Wirth: From Sheremetyevo to Domodedovo.

Trichina: From northwest to southeast… [laughs]

Wirth: Right.

Trichina: …of Moscow, which probably makes 80 kilometers if not over 100.

Wirth: Yeah. It took a long time. Passed the Lubyanka. I just changed and went. We flew at night. All Russian airplanes seem to fly only at night.

Trichina: Long-distance?

Wirth: Yeah. To Samarkand in Uzbekistan. And there was a one-week conference there. It was really kind of an interesting experience. It was for young scientists specifically.

Trichina: Was it an international conference or conference with internationally invited guests?

Wirth: The latter. Yes. It was organized by the people in Novosibirsk. A place you know very well. You studied there.

Trichina: And I even worked there.

Wirth: You even worked there. Yes. Then afterwards, we flew to Tashkent, had a very nice lunch there in the home of the local party secretary. Then in the evening, we flew to Novosibirsk, again at night. And…

Trichina: Of course you fly at night and you arrive in the morning. This way you don't waste your time for work. [laughs]

Wirth: Right. Yes, yes. You're really sensible. Anyway, I stayed a week in Novosibirsk, and then away back via Moscow, where I voiced my wish to go by the Trans-Siberian Railway, not the tourist train but on that track. And I spent 50 hours in that compartment…

Trichina: In the train?

Wirth: …in the train, together with a single person who didn't know anything but Russian or didn't want to show that he did. Probably an apparatchik.

Trichina: At that time, you didn't speak a word in Russian?

Wirth: Not a single word. No, no.

Trichina: Yet you survived?

Wirth: Yes, yes.

Trichina: How did you survive? [laughs] Fifty hours is a long time.

Wirth: Well, the professor in Novosibirsk when I was on the steps into the car pulled out a bottle from his raincoat and handed it over to me, a bottle of vodka. And I thought, "Oh, what a silly idea." But then the dining car provided fairly lousy food, so I decided I'm there once only. And I survived first of all by buying apples and пулочка *[little buns]* from the ladies on the platforms on the various stations, and from the bottle of vodka. And it was empty when I arrived in Moscow.

Trichina: Well, anyways, it is not a bad diet. [laughs] You didn't get sick and you survived.

Wirth: Oh no.

Trichina: [laughs] Okay. Where we are now?

Wirth: Well, in Novosibirsk. And the fascinating thing for me there was to see a project which had been built there *[1:10:00]* by the name of Kronos, very much according to my Lilith project. And they had done a very good job. It was just a few…

Trichina: Very young people.

Wirth: …three or four young people. Yeah, yeah. So they had had the complete freedom of pursuing such an esoteric project, and it was very touching.

Trichina: Did you speak with them?

Wirth: Yes, of course.

Trichina: And they spoke English?

Wirth: They spoke English, yeah. More or less, yeah.

Trichina: [laughs] You could understand it?

Wirth: Right.

Trichina: Right. I don't know what actually became of Kronos, the whole project. I suppose after *perestroika* time, there was no funding for some time, so probably it sort of…

Wirth: That's right. They spread everywhere, yes.

Trichina: Okay. Of course your professional life, professional interests didn't finish with your Turing Award, huh?

Wirth: Not quite, no.

Trichina: It continued, so it was Oberon and Ceres. And now we come to turn around, and from software via software–hardware back to hardware, FPGAs and…

Wirth: Yes, yes. The mixture between soft- and hardware. Yeah, yeah.

Trichina: [laughs] What prompted your interest in FPGAs?

Wirth: Well, the first FPGA came out in the mid '80s I believe by a company, a very small company, Xilinx. I, as many others, thought it was a rather crazy idea, a big waste of resources and transistors. But we decided to take up some work and build even some editing and routing software, even a language called Lola, logic language. That was all in the '90s.

Trichina: What was the goal of this project? Was it like teaching projects?

Wirth: Yes, mostly. Also, at that time, in the early '90s, I got wind of a project that was going on in the mechanical engineering department about building a model helicopter that was autonomously flying, where essentially you could pre-program the computer with a route that it would fly. I looked at that and saw immediately that they had overdone it. They had two Intel 486 processors which consumed a lot of power, and I was absolutely convinced that's not necessary. And I decided to build my own computer for them. It's actually right here. That's the second version. The first version is a much bigger board. But that was the thing that was flying on the model helicopter. Some processor here, a

StrongARM from DEC, an engineering sample, memory, and ROMs. And the rest is convertors, power supplies, and mainly connectors.

Trichina: What about sensors?

Wirth: The sensors are here. One…

Trichina: But we don't see them. [laughs]

Wirth: Yeah, these small chips, it's all inside, you know?

Trichina: A good piece of engineering. Did it actually fly?

Wirth: Oh yes. Yes, yes, yes.

Trichina: It did?

Wirth: It did, yes.

Trichina: So when was it? Which year exactly when it went to the sky?

Wirth: Probably about 1996-7, the last version.

Trichina: Okay. So a full 20 years before the era of drones as we have it now.

Wirth: Yes. Oh yes. But the Swiss Department of Defense never showed any interest.

Trichina: 1996, it was also the year of your second trip to Russia, to Novosibirsk. This is where we met first time. I remember it was at this conference, yes. [laughs]

Wirth: That is true. Yes, yes. I guess we danced.

Trichina: We danced, indeed. [laughs] Do you want to say a few words about this second trip, your impressions with respect vis-à-vis the first trip? It was six years later.

Wirth: I believe it was the… my big trip into Russia, wasn't it?

Trichina: Yes, I think it was a big trip into Russia.

Wirth: It started in Petersburg and then Moscow, a conference, and then…

Trichina: Saint Petersburg again, Petergof …

Wirth: No, that was beginning.

Trichina: Pavlovsk

Wirth: No, but we went to Monino to see the large open-air museum of the Soviet Air Force.

Trichina: Was it impressive?

Wirth: Very, very. I was overwhelmed. It was of course my specialty, you know?

Trichina: So you knew all the airplanes by heart?

Wirth: No. The Russian ones, they were not so much in publicity. But I was very impressed. They have huge things.

Trichina: Who was your guide there? How did you get there in the first place? [laughs]

Wirth: [laughs] Well, I have my connections.

Trichina: [laughs]

Wirth: No, no. It was arranged by some people from the computing profession, and I stayed a whole week in Moscow and one day we went there to Monino, about 80 kilometers east of Moscow. And one day we went to Yegoryevsk. And there I met with a colonel and his group. They showed me their drones. That was also very interesting. In the countryside, Yegoryevsk. And we went for lunch and had a party with vodka of course.

Trichina: [laughs] Well, some things don't change with *perestroika*. So now we're approaching your retirement time. Or you want to say something else about your work in ETH?

Wirth: Well, of course the time between my retirement, the 10 years, were full of trips. I had the time.

Trichina: You mean *after* the retirement?

Wirth: After the retirement, yeah. The Russian trips were a part of it. There were three to China also, but… They're very impressive, but of a different nature. In Russia, I got to know people quite personally even. In China, less so. In Russia, I got up to Novosibirsk and even further, to Tomsk. And there they told me… I thought I had been really very, very… travelled very far. Then they told me, "Now you're in the middle of Russia." [laughs]

Trichina: Exactly. Geographically, yes. [laughs] Exactly.

Wirth: But China was also something. I mean the first visit to China was I felt to be in a different continent, and on the second one, again, different planet. Because in those 20 years, China had completely changed, and I think now it would be again the same thing. But that was very, very revealing to me, and I realized for 70 or 80 years, we here had been so exclusively oriented towards the West – France, Britain, America – that we knew nothing about the East so to say. We knew that there is this dangerous Kremlin, but not aware that there are 150 or more million people around it, and these people are like you and I. Not at all any war mongers or dangerous. And I really felt very relieved when Gorbachev came to power because it seemed to be a slow opening.

Trichina: Well, the curtain was on both sides, if I may say so. [laughs]

Wirth: Of course. Of course. This is also something I realized, yeah. We had lots of misconceptions, and I'm sure Russians have them too.

Trichina: Did the fact that you started to learn Russian after your retirement, did it help you to communicate with people, to understand a little bit more about this new…

Wirth: Yes, absolutely.

Trichina: …but not so different world at the end, I mean a human world?

Wirth: Still on the same planet.

Trichina: Yes. [laughs]

Wirth: Yeah. No, no. That helped and that was of course my hope. Well, I wanted, I had this academic goal to test whether after retirement *[1:20:00]* I'm still capable of learning something. But it worked out okay. But after a while, the energy dropped and…

Trichina: Hmm. But I suppose you still can read Russian, right?

Wirth: Oh yes. And I can even write Russian.

Trichina: And you can even write Russian. Speaking of course without oral of practice is much more difficult.

Wirth: And understanding, that's the most difficult.

Trichina: Well… [laughs]

Wirth: Because the Russians speak so blurred.

Trichina: Uh, I think just the Russian vocabulary is so rich and we have a way to say the same thing in so many…

Wirth: And they glue words together. Yeah.

Trichina: Right. So where we are now? We are now after the retirement. Learning Russian was one intellectual exercise. Maybe also emotional, because for example I noticed in your bookshelf quite many books of Chekhov.

Wirth: Yes, yes. Yes, yes.

Trichina: Apparently he became one of your favorite authors.

Wirth: Right. He writes without undue complexity.

Trichina: Exactly. And when I read your letters in Russian, I thought actually your style is very similar to the style of Chekhov.

Wirth: On a modest scale, yes.

Trichina: On a modest scale. But also you're still engaged in what one can say professional activity in the further development of Oberon System and porting it. Would you elaborate on it so that people know what is happening?

Wirth: The incentive came from colleagues mentioning that they had learned very, very much from my book called *Project Oberon*, which is one of the very few describe a full system and its development in detail. That's something in computer science that is wrong. People learn to write before they have learned reading. Anyway, the book is there, but it got very quickly out of print. It was too specialized and narrow clientele. And they said, "Well, that's really just too bad. Couldn't you make a revision, a second edition?" And first, I went, "Well, no, no. It's old hat." But then when I started thinking about it seriously, I thought, "Well, it might be a good idea."

But then one large chapter, the largest chapter in that book was on compiler construction. That was for a processor that we used in Ceres which didn't exist anymore. Vanished from the surface of the Earth. Because it was too complicated.

Trichina: [laughs]

Wirth: Really. Really! So I thought, "Okay. Choose another processor, write another compiler? No, no, no, no, no. Then use this change to design your own processor, again according to the principles of regularity and simplicity." Then

that friend told me, "Okay. That's a great idea. You can even implement it. Now we have these field-programmable logic arrays. You can even implement it." And that's what we did. Now we have commercial development boards about this size here, from Xilinx, and we have implemented my own processor and the software is there, almost. And we have many people who are interested in the whole world.

Trichina: So it is all open-source?

Wirth: It's all open-source. It's on the web.

Trichina: And one can find on the web? So probably going to your website, and then from there going to "Oberon Project."

Wirth: Yes. It's called "Project Oberon."

Trichina: "Project Oberon." What about your RISC5 experience?

Wirth: That's it.

Trichina: That's it? So…

Wirth: Yeah. The processor is called RISC5.

Trichina: And is it reliable? That your design is stable?

Wirth: Yeah, yeah.

Trichina: Can be used?

Wirth: Yes, absolutely.

Trichina: People report your bugs?

Wirth: Recently, a bug was reported in the multiplier code.

Trichina: Cannot be. [laughs]

Wirth: No, I thought so too. It's actually in that part which is just when you ask for unsigned multiplication, which nobody ever uses. But the signed bit was treated in a wrong way, and I fixed that over the weekend, and so all settled. It's all open-source. I understand it.

Trichina: Niklaus, we spoke about many projects that you undertook in a span of more than 40 years of your professional career. If I ask you to tell me a story that illustrates the highest or the lowest point, what would it be?

Wirth: That's a bit difficult to say. You don't necessarily mean professional experiences in general?

Trichina: Well, here I meant professional experience.

Wirth: Well, it was of course when a system came out and was presented successfully, so to say the termination of a certain development period, that was always very satisfactory, very satisfying.

Trichina: It was not the moment when you were awarded the Turing Award the highest point? You don't think so?

Wirth: At least I don't remember it that way. I'm not the type who attributes too much significance to these events. But it was certainly an important recognition, and that is always welcomed, that's for sure. Particularly after time, you have periods of time when you feel there is not much recognition coming in, and then such a thing is very gratifying.

Trichina: And the lowest point?

Wirth: The lowest point. Oh, I forgot them.

Trichina: [laughs] Good, good. Which means I… you remember the highest point.

Wirth: Yes, yes.

Trichina: Or in general, you remember the process, the satisfaction from the process, the elation when the results are achieved. How many PhD students did you have altogether during these 40 years?

Wirth: Only about two dozen over these almost 40 years.

Trichina: And where they are now? All over the globe? [laughs]

Wirth: Well, yeah. I lost touch with most of them. But in the last 15 years of my career, practically all of them to went California, where I could pick up a job for them with high-caliber research people and research establishments. And many are still there.

Trichina: But some of them came back?

Wirth: Few.

Trichina: Few?

Wirth: Yeah.

Trichina: We spoke a lot about work. What about life? The hobbies? One you mentioned at length actually, the airplanes.

Wirth: Yes.

Trichina: Languages? Reading books in three or four languages? At least German, English, Russian.

Wirth: Yes.

Trichina: French?

Wirth: French?

Trichina: Maupassant? I saw Maupassant on your bookshelves. [laughs]

Wirth: Yeah, that's true.

Trichina: What else?

Wirth: Well, the hobby activities, first of all working in the electronic shop at home, that was a pleasure. Mostly, not always. And I liked to go hiking in the nature, in the mountains even. But nowadays at my age you don't do that so professionally anymore. No.

Trichina: But you can still enjoy it.

Wirth: I can still enjoy it, yes, yes.

Trichina: Once you mentioned – I don't remember where it happened – your flight on a fighter jet. Can you tell me about this story?

Wirth: Yeah. That was of course an incredibly big experience which moved me a lot. It was a friend I knew in America, and I could go and fly in his Tiger F-5 fighter jet.

Trichina: A private fighter jet?

Wirth: Yes. Yeah, yeah. He got rather rich through Microsoft.

Trichina: Okay. But still it is quite amazing. [laughs]

Wirth: He did not only have this fighter jet. He had other commercial jets and he had two helicopters. *[1:30:00]* Anyway, yes. And that was an experience of a lifetime, I tell you. I mean in the last 10-15 years when I had trips into the US, I always took some flying lessons. That's very much easier in America than it is here, and much cheaper too. So I learned flying. But sitting in a fighter jet is a different kind of game, very much more serious business.

Trichina: Was it scary?

Wirth: Depends how you take it.

Trichina: I mean with such a speed. [laughs]

Wirth: Yeah, yeah, yeah, yeah. The speed wasn't sc-… Once you sit in there, it's okay. But after takeoff, he said the airplane is mine, and I did the controlling and everything. And I flew some loopings and rolls. So that was very exciting. When you are on top, you look up like this and you see the Earth, you know?

Trichina: [laughs]

Wirth: No, it was… you go up to a thousand kilometers per hour, but you don't feel that much. Except you are of course completely surrounded by glass.

Trichina: Something to tell about your grand-grandson, newborn grand-grandson. [laughs]

Wirth: Yes, yes. It will take a few years.

Trichina: Okay. I think now it's time to wrap up. Of course, you wouldn't pretend to be a prophet, right. But still you think… I suppose you think and you thought a lot about where programming as a profession is headed, where computer science is heading. Can you share your insights and your maybe doubts, misgivings?

Wirth: Let's say it from the start that in general the situation is disappointing. We still had times when there was science in our subject. Also, the school has completely transformed itself. I don't say it was wonderful when I was a boy, a student, but at least it was a school and not an industrial complex. Many things are much better now than they were. It's certainly much bigger and much better funded. But in computer science, I think too that too much is being driven by business, by research grants getting the money to do things, and less by the sheer desire to do science and to create new knowledge. I mean that's essentially what university life is primarily concerned – to create and disperse new knowledge. And nowadays it's mostly business.

Trichina: Well, maybe without business aspect, it is hard to expect huge progress, because at the end the investments are done in something that can again in turn bring money.

Wirth: Absolutely. That's right.

Trichina: So don't you think that you were living in a sort of glass castle and where you could…?

Wirth: In an ivory tower, yes. But the great scientists, start with Pascal and Leibniz and so on, Newton, they had no business in mind. And of course it would be silly to think things can go on like that now, but I'm just saying the tendency is a little bit too strong to only count money. It's of course a reflection of the entire world where there's only one thing that counts – power, and that is money.

About programming…

Trichina: And programming languages.

Wirth: Yeah. Yes.

Trichina: What would be their role?

Wirth: Maybe I should repeat that, that my idea was that programming languages allow programming on a higher level of abstraction compared to machine coding. You can abstract specific properties and facilities of a specific machine. You can abstract to a higher level and create programs that will then be available and runnable on all computers. That's called abstraction. And the term "higher-level language" comes exactly from that.

Now to create the general-purpose language which relies only on higher-level concepts is not easy. I mean think of it. You cannot make any reference to any machine anymore. You have an abstract thing. You may have created a model for an abstract machine, but that's all. You cannot go to lab and say, "Well, let's try it. Let's see whether it works." That's out. But unless you strive for that concept of machine-independence – "portability" is the same thing – you'll never really make great progress in the art of programming.

Look at today's situation. People program in C++, the worst disease ever created. Or C# or Java, which are a bit better. But they all suffer from their mightiness. I'm always expecting they're going to collapse under their own weight.

Trichina: Well, it is interesting that some languages win no matter what are their theoretical merits, and some languages lose if even theoretically and even as a user, for a programmer they would present a really good tool of abstraction. How

you can explain it? And apparently it is not only because industry picks up the wrong languages. For some reason it picks up those languages. How can you explain this situation?

Wirth: It's not easy to explain it in full, but I mean look at the current situation where Java and C# are predominant. It's simply because of that feedback. Industries create these languages, then compiler to propagate them. There are users who flock to them. There are schools which start teaching it. And this way they spread. Then soon, when software projects are announced, it is requested that you know one of these popular languages. And for other languages, you couldn't get anyone. So it's a lot of feedback processes which go to that. It was broken only once, with Pascal. That was kind of a miracle. And I kind of knew that it would never happen again. But Oberon would really deserve it at least in the domain of teaching, because that's where it excels in my opinion.

Trichina: But you mentioned yourself that universities are becoming more and more grant-driven, money-driven. Therefore they do what industry wants them to do, maybe implicitly however.

Wirth: Exactly. But already in 1970 with Pascal, I realized that industry asks from the university to teach what they use, and universities are teaching what the industry requests. So again a tight feedback loop. Dijkstra once said, "You should not give industry what it wants but what it needs." It's of course a little bit arrogant, but there's a lot of truth in it.

Trichina: Well, Dijkstra had this ability…

Wirth: Absolutely.

Trichina: …to say good things, right things, while maybe in a little bit too-straightforward way to the gist, but…

Wirth: Yeah. Sometimes he edged on.

Trichina: Yeah, well… Well, I think our interview comes to the end, and I want to ask the last question. Partly you answered already this question. What was the unifying theme *[1:40:00]* of all your projects?

Wirth: Well, the unifying theme, if I think of it, is probably "Make things as regular and as well-structured and as simple as they can be without losing sight of their purpose." Undue complexity is a bad thing. And it's unnecessary. Except in the modern commercial world, it's often used to hide inadequacies. If something is intransparent, it's much harder to copy it.

Trichina: [laughs]

Wirth: So not all these bad things happen just by chance.

Trichina: Obfuscation by design? [laughs]

Wirth: Right, right.

Trichina: May be, may be.

Wirth: At least I would say universities should have been more keen on proper tools, proper languages in our case. But it wasn't the case because, as I said, universities want to teach what is requested from them.

Trichina: So our interview came to the end, and what is left is just to thank Professor Niklaus Wirth for his time, for his openness. And I hope this interview would be interesting not only for contemporaries but for future historians of science. Thank you, Niklaus.

Wirth: Well, thank you, Elena, for agreeing to conduct this interview with me. I'm very happy to have chosen you. You were ideally suited. You know my work and you know me personally to some degree at least, although we speak different languages and live in different places. But I enjoyed it and you did an excellent job. Thank you very much.

[end of recording]