



## **Oral History of Charles Simonyi**

Interviewed by:  
Grady Booch

Recorded: February 6, 2008  
Mountain View, California

CHM Reference number: X4428.2008

© 2008 Computer History Museum

**Grady Booch:** I'm here with Charles Simonyi [February 6, 2008]. Charles, it's just a delight to have you here. You have had an extraordinary life, and we're here to capture an oral history of your life thus far.

**Charles Simonyi:** <laughs>

**Booch:** My goodness, you grew up in the time before there were really any kind of meaningful computers to the time when you've gone to space where your life depended upon those computers working properly. So we'll try to fill in all the details in between.

**Simonyi:** <laughs>

**Booch:** So you grew up in Budapest. You were born there. Tell me a little bit about your family.

**Simonyi:** My dad was a professor of electrical engineering, and in fact he was teaching a course, Theoretical Electrical Engineering, which practically every engineer in Hungary had to take. So still when I meet somebody Hungarian they usually refer to the fact that they either learned from my dad's book, or actually took classes from my dad.

**Booch:** What was the name of that book? Do you remember? Do you still have a copy?

**Simonyi:** Oh, of course, he wrote a number of books and obviously I have a copy. I never had to learn the theory of electrical engineering, which at that time was really Maxwell's equations. So I know Maxwell's equations by and large, mostly because the students usually gave a gift to the professor at the end of the year, and this gift was something like an ashtray or some trinket, but always with the Maxwell's equations engraved on it. And so I'm familiar with them, but I'm familiar with them in the German notation, not the English notation, you know, for example "curl," what we call "curl" is called "rotation" in German, so it's "R-O-T," instead of "curl".

**Booch:** Right. In what university was he teaching at that time?

**Simonyi:** Well, it's the University of Budapest, the Technical University of Budapest. It's the best university in Hungary.

**Booch:** And that was in 1948 when you were born. Any siblings that you had?

**Simonyi:** Yes, I have a younger brother and we see each other all the time.

**Booch:** Tell me a little bit about Budapest in the late forties. What was it like living there?

**Simonyi:** Well, I think that I can't talk about the forties, because I was born in '48 <laughs>. However my earliest memories are in '53, '54, that timeframe. I remember for example, when Stalin died. That was a big deal. I mean I knew nothing about Stalin, and death, for that matter, but there was the incredible excitement, or foreboding, or a mixture of the two. I think it was a-- but I was a little bit strange too, so I can't calibrate it quite. But I remember for example, when the 1956 revolution came. I remember it fairly clearly. I was in second grade, and I was enjoying it tremendously for two reasons. One of them is that we didn't have to go to school, and the other one was that we stayed in the basement, so it was kind of camping out. We didn't do many adventures with the family. So living in a basement and camping there was in a way an adventure. And I always felt that war to me seemed like a season, and it seemed very natural that it would come at some point, so it didn't seem unexpected. The second world war was still fairly recent, and certainly the signs of war were all around you. Then later when I left to Denmark, my first impression was that the buildings didn't have bullet holes in them. That was the first time I actually saw a building, other than a brand new building say, but a building that didn't have some bullet holes some place. It was very typical in Budapest that everything had some damage, and furthermore, nothing was repaired. So the damage of the war <laughs> was clearly visible, and of course after the revolution there was an additional layer of damage. There were bridges, I remember, across the Danube that were blown and still not cleaned up.

**Booch:** So my history is obscure, when was the revolution, what year would that have been?

**Simonyi:** '56.

**Booch:** Okay, and that was really around the time you were discovering computers.

**Simonyi:** Ah no, no, no, let's not exaggerate. I discovered computers in about '64, I would say, but the lovely thing is that Hungary was in a time warp, so that the computers I discovered are the equivalent of the computers that were used in Russia quite a bit earlier. Maybe they were hand-me-downs, from the Soviet Union, the computers that I saw and then the first computer that I learned. So these were older generation computers, in fact, the typical first generation computer. And in that way, even though it happened fairly late because of this time warp, I managed to learn about computers of a much earlier era.

**Booch:** In fact your first computer, I recall, you worked on was the Ural, is that the name of it?

**Simonyi:** That's right, the Ural-2, not the Ural-1, which was a simpler serial machine. The Ural-2 was actually a wonderful, wonderful machine that was first generation with vacuum tubes, but it had core memory and it was a parallel machine with floating point.

**Booch:** Tell me about your first encounter with it. Where did this machine show up, was it at the university?

**Simonyi:** Well, what really happened was is that I went to a trade fair and I saw some more modern computers, just caught a glimpse of them. I saw the data sheets on these computers, and I was really disappointed in that looking at the machine I couldn't make heads or tails of what they were about, and

what the data sheet meant. They were talking about words and cycles and I couldn't imagine what that was or why this box would be useful. And I kind of agree with Don Norman, that probably with modern technologies you can't decide what things do by looking at them. I always loved the mechanical devices, for whatever purpose, because they might have looked complicated but if you looked at it for long enough and played with them long enough you could figure out what they were for. These machines seemed completely different, and it peaked my interest, and I couldn't find any literature or any source of information. I bothered my dad and he was a theorist, so he knew nothing about computers. He knew that flip-flops were important, and he could tell me about the theoretical aspects of flip-flop design. But he pointed me to one of his very talented students, Zoltan and I hooked up with Zoltan. Zoltan was the chief engineer at this Ural-2 that was at the Central Statistical Bureau. I visited this machine once and I was blown away by the grandeur of it. It's bigger than anything that I've seen before. It's as big as a 6600, and it certainly has more lights. The tube machines had the great advantage that it was very easy to put one of those little orange gas discharge lamps on every flip-flop. It was just part of the flip-flop. The tube was sticking out from a unit, and next to it there was a light that was indicating the state of the flip-flop, and the tubes were of course glowing constantly in a very nice orange color, and then the little light was blinking. And if you can think of a thousand of these in glass cabinets, or rather, cabinets with a window in the doors, the result is amazing. And of course as the machine was running the whole room was flashing in intricate patterns, very much like the cheap science fiction movies. I'm sure that's where they got the ideas from.

**Booch:** You would've been 14 or 15 around this time, when you first encountered it?

**Simonyi:** Yeah, I was about fourteen, and I was lucky in that I was very determined and also, I don't know, impertinent, in that I refused to believe that this was anything beyond understanding, and in fact now we know that it's actually much simpler than differential calculus. But at that time that wasn't clear. You were supposed to be a PhD or somebody to touch it or to play with it, or work with it. And in fact, I remember a little bit later I was at the console of the machine and the director came in by chance and said "What is that kid doing there, he's going to break the machine?" It was obvious to everyone that if something's expensive and then you touch it the wrong way then it would break, which is typically true for mechanical things, you know, without interlocks. It's obviously not true for a computer. But I gained understanding of what was going on, and that gave me the confidence that I can actually do something.

**Booch:** So tell me some memories of programming the Ural, what kinds of things did you do, and what is the experience of programming?

**Simonyi:** <laughs> Well, there's so many interesting things about that, well, at least to me, interesting things. One of them is that the world per se didn't give you any hints of what to do with it. The Statistical Bureau was using the computer in a very awkward way. I think they were hampered by the communist ideology, so they were using it for planning the economy. In particular they thought that they would be able to optimize the distribution of cement. And so they were really obsessed by cement shipping, or just cement in general. And of course it was complete nonsense because there was no way to get the data in time, and also what do you do with it. And the problem with cement certainly wasn't that it wasn't distributed, the problem was that it wasn't available, and they could've just created a, you know, let the price float and it would have been solved in no time, but that was politically incorrect. So, they had to do something scientific about it, and I mean it was completely hopeless. The machine being completely

underpowered, the data not being available, and the purpose being complete nonsense. So those three things working together, that wasn't interesting.

So I had to view the machine as somewhat of a toy, and in fact my first program that I ever wrote, I mean I was searching it and my soul of do I know of any algorithms. I mean I got the idea that there is a thing called "algorithm" and do I know any algorithms? And the only algorithm I could think of is a parlor trick where you write down a sequence of numbers in a odd rectangle, like a five by five rectangle, so that it becomes a magic square. And you don't have to remember, you know, of course you could memorize a magic square and then say, "Look guys, here's a magic square," that would be impressive but boring. But if you remember an algorithm of how to write the 1, 2, 3, 4, 5, 6, 7, 8. There's a very simple algorithm that I saw in a book of magic, and that creates a magic square. It generates a magic square. And so I can make a seven by seven magic square if you want, it just takes a little bit longer time. I don't have to memorize it, and I can start at a different number and it's still a magic square. A magic square plus a constant is also a magic square. So you start at-- let's start at-- you tell me how big a magic square you want, where do you want to start, and I write it down, very impressive. So I put that into the-- that was my first program. I had the machine all to myself for a Saturday afternoon, and when everybody went home and I worked from noon until 8pm and I got the most terrible headache at the end, but I had a 81 by 81, or some ridiculous size magic square <laughs> at the end of the day, and I overcame so many problems.

Oh, by the way, this happened after I talked to Zoltan, and he described to me the instructions, and the instructions were numbered by octal op codes, so 01 was addition, I mean 00 was no op, 01 was addition, 02 was clear and add, 03 was subtract, 04 was unassigned, 05 was unassigned, 06 was multiply, 07 was divide, 10 was transfer sign, 11 was shift, and so on and so forth. And I learned this in one afternoon. There were about thirty of the 6 bit op codes that were assigned. The next day I had to ask him a question about two of them that were a little bit unclear and were a little bit more complicated than the others, then I thought that now I can program. But unfortunately when I did this algorithm I found out that I cannot multiply, I mean I knew that 06 was multiply but it was a fixed point multiply, and I needed an integer multiply. So in a fixed point multiply you would multiply one-half by one-half, it would be one-fourth and so on. So, you could make an integer multiply out of fixed point multiply simply by shifting left by the word length the double precision result and you would get it. But that was beyond my understanding, so I realized that the multiply just didn't work for me, and I didn't know what to do with it. So I wrote a little multiply routine <laughs> by repeated addition, because I knew about loops, so I did all the index calculation in my algorithm by repeated addition and a loop. So the program ran a little bit slow <laughs>. But it was okay, the biggest one at the end of the day was like five minutes to do a really big magic square, but that was my first program.

**Booch:** And how did it feel to get that first program?

**Simonyi:** Well, I just remember the headache. Yeah, I mean the process was amazing. And now the key thing, and I took that feeling with me all the way. That it was a personal computer. That machine was all mine. I was sitting at a console, I had the-- I mean it's a giant console, it's about this wide, and in front of you there's a display about that big that had the lights, the accumulator, the register, index register, the instruction counter, all in ones and zero lights, separate, and grouped into wonderful groups of three so you could read out the octal number right away. I mean I looked at the-- I could see 16 immediately in the instruction register that is the store instruction for example, and the input was octal input. It was the most sophisticated input device I've seen or, you know, binary input device on a

computer I've ever seen, it was very much like a cash register, where you have a column for each-- for each position you have a column of keys. So if you had like we had forty bits thirteen positions plus sign, there were thirteen columns of digits between 0 and 7, so you could go like 1, 6, 4, 0, 3, 3, and if you made a mistake then you just changed that column, you didn't have to redo the whole thing. And then when you had the correct number you entered it into a register, or you started by clearing the keyboard, like in a cash register, you push the clear button that pulled that giant electromagnet underneath that released all the springs, and it returned to the original state with a terrific clang. And it was a very efficient way of, you know, you could basically punch in your program one instruction at a time, directly into the memory. And that was my input device. My input device wasn't offline doing some punch cards. We didn't have punch cards. We don't need no stinking punch cards. Plus we couldn't afford punch cards <laughs>. We could afford to have the machine, at least I could.

But this notion of having my own machine, I mean that stayed with me, and of course as I went through my career. I went to Berkeley. I worked with big batch machines. I was still a systems programmer, so when I worked on a machine I still had total access. I was never in the users' queue putting in some batch program and having it returned two hours later from the printer. So I always thought of computers, from the very first one, I could think of them as personal computers. And later when we get to the Apples, or whatever, I say "I know this machine, this machine is personal and it has the same performance and the same capacity as the machine, the Ural-2." And we are talking about 12K bytes, and so just to be a number, it was 4K of 20 bits, so call it roughly 12K bytes, and the floating multiply took 700 microseconds. So it's about, again, that's what a 6502 could do.

**Booch:** So when you had that afternoon of programming and you unveiled your results to the other folks in that center, what was their reaction?

**Simonyi:** <laughs> I don't think I was-- they were laughing or the-- but obviously that wasn't going to help me too much. But then I decided that what we need is a compiler, there were no high level languages, there wasn't even an assembler for the machine. So immediately the next thing had to be a compiler, so I started to work on a compiler <laughs> and then people were pretty impressed by that and that went on for about a year, as I was in high school. And then they realized that there's a kid that will help them do their job for free, and that gave me access to the machine. So I looked at it, I certainly wasn't interested in making any money, since money had no value, Hungarian money, and not convertible, so I looked at it as an investment.

**Booch:** And so you did this for a couple of years while you were in high school then?

**Simonyi:** About just one year, then I met my next computer--

**Booch:** Which was?

**Simonyi:** The Danish computer, GIER. So the Central Statistical Office, I guess, had enough of the Russian machine and they were going to buy this Danish computer, the GIER, which is a very typical second generation machine. So now after about a year we just went one generation of computers. GIER is an amazing, again, it's an amazing machine, and it's only recently that I realized that the Ural and the

GIER are actually related, they are both related to the Institute of Advanced Studies, the IAS, the von Neumann computer, the plans for which were published. So, and many copies of the IAS computer were built around the world, the Johnniac, the Illiac-- the first Illiac, and in the Soviet Union the BESM, the B-E-S-M, Bolshaya Elektronno-Schetnaya Machine I guess, was made according to the plans of the IAS computer. And then the Ural was a future copy of BESM, so it was the grandchild of the IAS machine. Similarly the Swedes did a copy that was called BESK, B-E-S-K. And then the Danes made a copy of the Swedish machine, which was called DESK, D-E-S-K. And GIER was the improvement of the DESK, so that was a grand-grandchild of the same machine. So these guys were kind of a grand uncle of one another, which had I actually studied and now knowing the familiar relationship I can see the similarities in the architecture, they are all corrupted by the small modifications and improvements that were done at the time, but many of the key points are still noticeable, which is amazing. 26:50

So anyway, there was this machine, the GIER, an amazing machine, the first one with an Algol 60 compiler of the highest caliber, the best implementation, this machine had a 1K memory, 1K core memory, but 42 bits, and it had a drum memory of about 16K words. So again, so it's about 60 kilobytes of drum, and 5 kilobytes of RAM, but faster than the Ural. A floating multiply might be 200 microseconds. It was micro programmed. It was very, very elegant. And it was surrounded by incredible software people, which was completely new. In the Ural there were no software people, it was all electrical engineers. The machine came with a square root and a sine routine, but it didn't come with any software whatsoever. And even the sine routine, and I'm talking about the trigonometric routine, came on a technical drawing, where the instructions were hand filled in into a mechanical drawing of how to compute sines on this machine.

**Booch:** So you taught yourself Algol?

**Simonyi:** Well, Algol, that's right, well Algol wasn't that hard. What was the key there is that I got hold of the Algol compiler distribution tapes.. There was also a couple of papers that were published by Peter Naur and Jørn Jensen, that I've read assiduously, and with not too much English understanding, and especially I had no understanding of technical English, and I remember a phrase, there was a point where it was discussing the youngest incarnation of the lexical graphically enclosing block. So <laughs>, of course, you go into a dictionary and you try to look up "lexical graphically," or something like that <laughs>, or what is this "incarnation" <laughs>. So the lexical graphically enclosing block, that didn't make much sense for a long time, and actually it was a stack reference, or sorry, it was a stack reference to the youngest incarnation of the lexical graphically enclosing block. So the dictionaries didn't help at all <laughs>. I mean starting with the stack, your reference is okay but you can imagine, so it was clear that I would have to learn this from scratch, or figure out for myself, and I had the help of these distribution tapes which were the assembly language tapes with all the comments and all the white space removed <laughs>, essentially for compression purposes. So it was a paper tape, even then the paper tape was about this round (spreads hands about 10"), an eight level paper tape, and it was a full Algol compiler but with all the white space and all the comments removed. So I had to this listing <laughs>. Which by the way, listing, that was a new thing. On the Ural, we didn't have a printer, so I never had a printout of my compiler for example, I had the chicken tracks on paper, that was my compiler, and it was written octal absolute, so there was no source. It's only recently that from the old binary tapes, I could convert it into an image and then print it out, and had the first listing of my compiler <laughs>. But, so, but with the GIER of course we had a normal printer, I mean, the high speed printer, amazing stuff, so I had this printout, without comments and without white space, and it's amazing what young kids can do when they have a lot of time and I figured a lot out <laughs>. Not everything, but I figured a lot out from that.

**Booch:** And so you spent a couple more years again with that computer, did you not?

**Simonyi:** That's right, no, about one year, one year. And then came a fateful day when an engineer came to repair the machine. Of course my plan all along was to go to the West. I didn't know how I would do it, when I would do it, but it was, yeah, I saw it as a necessity, that I saw no future in Hungary. I was very, very pessimistic about what I would be doing and I thought that the whole thing was completely crazy. And of course when I saw the, you know, when I looked at the Soviet technology, and then we acquired this paper tape reader that was done in Sweden with American components, so that's the first time I saw resistors, for example, with the color coding. Remember resistors with the--

**Booch:** The browns, the reds, yeah, all those colors, yeah.

**Simonyi:** Yeah, yeah, yeah, well, it was mnemonics, but I'm not going to say that because it's politically incorrect, but it's essentially numbers are, let's see, black, brown, red, orange, yellow, green, blue, violet, gray, white.

**Booch:** Impressive.

**Simonyi:** No, I'm using a mnemonic <laughs>.

**Booch:** Which you're not telling me what it is <laughs>. That's all right.

**Simonyi:** Everybody has different mnemonics. So, but just when I saw those components, those competitor components, I said "Wow, I'm going to waste my life if I have to work with stuff that is not as good as this other stuff" <laughs>. So anyway, so I was kind of trying to get out. This is now, I'm maybe what, fifteen, sixteen, and I start to speak English fairly well. It hasn't improved. I think my understanding improved quite a bit, my speech hasn't improved unfortunately. So I started to speak English very well, fairly well. Nobody around spoke English. An engineer came from Denmark to do maintenance on the computer. Of course I have to baby-sit the engineer because nobody can talk to him, so I'm still working around the computer with all my free time for free, and they tolerate me because I'm very useful, and so I can break all the rules. So I'm translating, or baby-sitting the engineer, and so at night I write a little demo program in Algol and the letter, and put it on a paper tape, eight level paper tape, on the GIER, and next day I take him to the toilet, show him where the toilet is, and I slip him the paper tape <laughs>, you know, "Take this to your leader." And this guy was traveling with a hundred paper tapes and paper tapes were this unknown quantity, unreadable, so a perfectly great way to encode a secret message. And he takes the paper tape, and also it's a very effective way of-- I not only had a written-- I didn't have a resume, I'm a high school kid, right, in Hungary, so I could present, you know, it's just that "Dear Sir, I'd like to work at RegneCentralen and I have a good understanding of the compiler," and blah-blah-blah-blah, "and here's a little demo program for your enjoyment." And then there's a demo program and you run it, and it's fun, it was executing or interpreting an arithmetic expression and explaining, as it was doing it, what it was doing. So fairly-- I mean super straightforward by today's standards, but at that time that was a big deal, you know, recursive, blah-blah-blah. So the guy takes it. A month later we get a telex that "Yeah, sure, come on, we need, come on out." Oh no, actually what happened is, "The director is coming to Budapest and he's going to meet you." The red light is blinking.



**Booch:** Yes, keep going, we've got a few minutes yet. <laughs>

**Simonyi:** Okay good. So, the director, Neils Everbek [ph?] came to Budapest and one day I get a call that he'd like to talk to me, and invites me to a hotel. That was the first time I was in a first class hotel in my life, in Budapest. And he had a lunch with me and on the spot offered me a job. Unfortunately he offered me a job not for the summer but for one year. And I said, "Well, that would be difficult, but we'll give it a try." And it wasn't my plan, but... And then it worked out, because I was underage and even though I was eligible for the draft at the age eighteen, as under age, well-- we put the military under a difficult-- well, it's kind of complicated to explain, but because I was underage nobody-- and I also got myself admitted to the university early on, nobody would believe that I would defect, or stay, defection is kind of the incorrect term, but that I would stay abroad illegally. Unfortunately because I was a kid I couldn't defect. I wish I could. That would've made the rest of my life much simpler. All you could do is stay abroad illegally.

**Booch:** Leave and not come back.

**Simonyi:** Not come back, and also, I mean expose yourself to all the vicissitudes of what happens on the other side. So, nobody believed that anything like this would happen, so they let me go.

**Booch:** And this would've been what year?

**Simonyi:** Ah, this was '66, I left Hungary '66, July 17<sup>th</sup>.

**Booch:** Wow, and let's take a break here so we can swap tapes.

**Simonyi:** Okay.

**Booch:** This is utterly fascinating.

**Simonyi:** <laughs>

**Simonyi:** So, on July 17th I arrived to Copenhagen. And I had a lot of fun by the way forty years later in 2006, July 17th, I retraced the exact trip on my own plane flying myself, but went on an exact, landed for example in East Berlin Schönefeld Airport, and Schönefeld Airport still exists. It's not used. It's kind of almost abandoned. The terminal building is kind of run over. I mean grass is growing and it's an abandoned building, but we found it and actually taxied out to the terminal building. We waited a little bit and then we replayed the whole thing. It was a big change in my life.

**Booch:** So you were 17, in Copenhagen.

**Simonyi:** At 17, arriving in Copenhagen on Sunday and 0.00 in my pocket. Fortunately two nice people from Regnecentralen were waiting for me and drove me downtown, and the director actually had lunch with me and maybe we just had a talk and then he pulled out 500 kroners and said, you know, "I understand you don't have any money?" And I said, "Yes sir," and he gave me 500 kroners, which was about a hundred dollars, a nice sum. Of course it was subtracted from my first salary, which is appropriately so, it wasn't a gift, and I certainly didn't expect any. And then next day I met my colleague, Per Brinch Hansen.

**Booch:** Really?

**Simonyi:** The Per Brinch Hansen.

**Booch:** The Per Brinch Hansen, how cool.

**Simonyi:** And Peter Kraft.

**Booch:** Wow.

**Simonyi:** Per Brinch Hansen just passed away last year.

**Booch:** I did not know that.

**Simonyi:** Yeah, unfortunately. Peter Kraft is still well and I visit with him all the time. But these two gentlemen, Per Brinch Hansen was a new PhD and was given the responsibility to do this new machine called the RC 4000 and do an operating system and do a process control system for the project for a Danish chemical company that was the main contractor. 41:34

**Booch:** Per would've been in his 20s around this time, would that be right?

**Simonyi:** That's right, just finished school, just straight out of school, and of course they were shocked, I mean this is a substantial project. A new computer, operating system, process control system, all to be delivered in one year and they need help and they get this Hungarian teenager with suspiciously long hair. So they were shocked to say the least. <laughs>

**Booch:** And this was your first paying job then, wasn't it?

**Simonyi:** That's absolutely right, absolutely right. This would be a paying job. I was completely on my own. I mean there was absolutely no way to support me from Hungary, simply because it's not that my parents had much money, but even the little that they had wasn't convertible. And of course that's why I came with 0.00 because it wouldn't have been convertible. So it worked out pretty well because I didn't

know that you were supposed to read manuals for awhile or get acquainted for awhile, so the first day on the job I said, "Show me the instructions and do you have input output conversion routines," and they said, "No, we don't." I said, "Well, we'll need them for sure, so let me write them." And I started on the first day writing code. I mean I really believed in them, I mean I found myself a job, I mean a task, I proved that it would be necessary. I mean I don't care about, you know, operating system or process control, but if you can't convert the numbers it won't go very far and we might as well do something. So they were pretty pleased. I got a raise after two months from, you know, whatever it was from \$200 a month to \$250 a month. <laughs>

And we finished up the project. It was an amazing, just an amazing project. It was delivered, probably the only time in history that the software was delivered before the hardware. We delivered in about, well not delivered but we were finished about a month or two before the hardware was really finished, so we spent about a month hand checking the code, just stepping through it and improving the comments and doing a little bit of simulation on the GEIR. In particular the process control portion was done with an interpreter just to make it compressed. We had 4K at that time 4K of 24 bits so 12K bytes of memory for this whole system. I had about 800 to 1,000 bytes to do all the process control equations that were quite complicated, but if you wrote the equations in ASCII, it would be about 800 bytes worth of equations, but that's enough because with a little 200 byte interpreter, you could interpret the equations, the machines were fast enough.

**Booch:** Do you have any stories about Per Brinch Hansen? I mean this is prior to the time when he was really doing his pioneering work on concurrency, wasn't it?

**Simonyi:** That's right, prior, that's right.

**Booch:** What kind of a man was he?

**Simonyi:** Oh, I think that he was a typical Dane with a pipe, he was very deliberate and spoke quietly and slowly. I remember once I walked into the room unseen by Peter and Per, and they were talking to each other and I noticed that they were talking in English, and I said, "Hey guys, I wasn't here, why are you speaking in English?" And they told me that it's easier for them, the two Danes, to discuss the system in English, since the comments are in English, the documentation is English. One good thing is that we grew up on basically the same information, which was the GIER Algol compiler. Per didn't work on the GIER Algol compiler much, but he worked on a Cobol compiler that was a followon on the GIER compiler. So he took the same background, the teachings of Peter Naur and Jørn Jensen, the two key contributors and he built on those concepts. Peter Naur believed that all data processing was essentially the same. There are some inputs and there are some outputs and there's some transformation in between, and that's how his compiler worked with multiple passes, but each with well-defined input and output languages and the focus was on what those intermediate language are. And we could apply the same principles pretty much to the operating system, so in the operating system too, the interface between the monitor and the user program was thought to be an intermediate language, and now we would call it an API. So he very much understood and grasped and followed that pattern, the pattern of compilers, but took it over to the operating system.

**Booch:** Now Peter shows up in your life again around this stage because he helped you find a position in the States from there.

**Simonyi:** Well, actually my dad had a contact at Berkeley, a professor, Cornelius Tobias, but Peter Naur gave the recommendation that got me admitted to UC Berkeley, which was just fantastic, because, you know, there would have been no practical way for me to take the SAT, so I was admitted under the weird quota.

**Booch:** But you were still flying under the radar because you were still a Hungarian citizen but illegally out of the country.

**Simonyi:** Yeah, yeah of course, oh that's right. Yeah, yeah, I was lying through my teeth. I mean I was, you know, I don't want to make too much of this, but growing up under a communist state, you got used to not being necessarily too candid about your circumstances or your thoughts for that matter. You typically said what needed to be said and then you thought what you thought, and unfortunately I had to continue that, just that mode of operation for awhile. But I'm sure now the statute of limitations has expired, so... but I wasn't always totally candid when people asked me something. I usually figured out what they wanted to hear and I answered what they wanted to hear. So, you know, I didn't discuss, for example, how long is my passport valid, and yes I did travel in my passport that has expired, but the expiration was printed in Hungarian, so I wasn't going to help anyone. And frankly I wasn't helped and I wasn't asking for help at any point, all I was asking for is people letting me interact with other people and do useful things.

**Booch:** So you then came to UC Berkeley, and that would've been what year?

**Simonyi:** That was in '67.

**Booch:** So wow, the peak. <laughs>

**Simonyi:** Wow, as we know, '67 was the summer of love. Unfortunately, I arrived at the autumn of love. <laughs> So yeah, I came about November of '67, which was, I'm not quite sure why I did it because I had a great life in Denmark, <laughs> I had a fantastic life as a matter of fact as by then an 18-year-old, you know, young lad with a fair amount of spendable income and a good job, you know, beautiful surroundings, fun friends, so but I thought that for my future I should go to a university in the States. So I had about \$1,500 saved during the year and I got admitted so I thought I would have to go to UC Berkeley, and I did and it was, <laughs> it was quite amazing. I mean I got into a situation where, I mean the biggest shock in Berkeley is this issue of work permit. I didn't make myself too familiar with the rules and regulations ahead of time. I mean I always figured that I have good skills and I will always find people whom I can help and then I can make a modest amount of money and do what I, you know, pay my way through life. And that was almost true except for this slight problem of work permit. Ahh, so, you know, I go to the computer center, of course they want a lot of stuff, they want a SNOBOL compiler, they want modification to their Algol compiler, I said, "Hey, I'm your man." And they say, "Oh, we'll pay you \$3.00 an hour." I said, "Wonderful." And then they say, "Oh, yeah, you're foreign student, let's see, do you have a work permit?" I say, "Work permit? What's that?" And of course it turns out that I can't get a work permit.

So I say to them, "Well, would you mind buying the program from me?" They said, "Sure, we won't mind." I said, "Great." <laughs> "So do you mind if I sit around here and, you know, just write a program at your desk?" And say, "Sure."

**Booch:** Very clever.

**Simonyi:** So that worked out pretty well because I mean it was way under the radar, it didn't reach, you know, the \$4,000 or whatever where I have to start paying taxes, but then I, of course, at some point I got a social security number and it turns out that it's a really good idea, I think it's the same thing with prostitutes, if you pay taxes, at least the IRS won't bother you.

**Booch:** Right, that's how Al Capone got finally trapped.

**Simonyi:** No, no, that he didn't pay taxes, right, no, but I'm saying that if you pay taxes, then they have to catch you for your other activities the other way, not through that.

**Booch:** Right, right.

**Simonyi:** So I certainly, well, I made \$4,000 the first year, so that was nothing, no tax event, but I was on that shadow economy where it wasn't entirely clear of what you can do, what you cannot do and so on. And I had to pay non-resident tuition that was \$400 a quarter, so I ran out of money very quickly. I had to check in with the INS, the Immigration and Naturalization Service, regularly and of course at some point I think I started to get a work permit. They could give a work permit if there were some unforeseen circumstances that made it necessary for you to work. And of course in my case the unforeseen circumstance was that I ran out of money, <laughs> ran out of the \$1,500 paying at \$400 a quarter non-resident tuition. So that was a completely unforeseen circumstance, so at some point I had to sit there with a straight face and say that sir, that was an unforeseen circumstance, I certainly did not foresee that. I mean it worked out and I'm eternally grateful for not being deported.

**Booch:** It was around this time you met Butler Lampson.

**Simonyi:** That's right. Butler wanted a, well, so much for that, enough of that, well, it's good to talk about that thing because it was overwhelming. I mean I had only two worries at the time, one of them is doing the programming and the other one is convincing people that it's an unforeseen circumstance, but you know, or just fighting deportation.

So yeah, I met Butler because Butler was the person who wanted a SNOBOL compiler for his class, and he put pressure on the computer center that you guys should have a SNOBOL compiler, because he wanted to teach his course in SNOBOL, which is a fantastic way of doing that. It's super, a great language for teaching especially. And I did a bang-up job of a SNOBOL compiler. I was amazed when I looked at the existing code on the CDC 6400, now this is a supercomputer, so now this is a third generation machine. So in three years I went from first generation to second generation to third

generation, a little over three years, but very, very quickly. I'm in a third generation machine, we are talking about 128K of 60 bits of memory, you know, multiple processors, well not as many, but still we had peripheral processors and subprocessors, not as much as the 6600 but it's still a formidable, formidable supercomputer. I looked at the existing code in the Algol compiler.

I remember the processing of the control card for the compiler, you know, where you're reading the parameters and there are about 20 parameters of setting different sizes or options or whatever from the control card. That code took more memory than the GIER had, so just by the reading of the control card, they use more memory than we ever used for all of the compilation on the GIER and that really shocked me. It turns out that the guy was using a macro instead of using a procedure and he was wasting...you say how can you use, well, we are talking about 5K bytes, how can you use 5K bytes for control card. Well, the answer is use a macro and you call it like 20 times and, you know, a macro for each option and you call it 20 times and it expands like 20 times. 5K bytes <snaps fingers>, gone just like that. It really hurt my, you know, my tender feelings <laughs> and so I did a great job with the SNOBOL compiler. It was the fastest, the smallest, because I was using the same techniques as we were using on much smaller machines, and now typically when people step, and it's absolutely true today, that people step up to a bigger machine, what they use is they use new techniques that are much more wasteful in terms of the use of resources, and there are some justifications for that, so I'm not, you know, I'm not an old fogey that doesn't understand what's beneficial and what's not, but at the same time, I mean now we have machines that are a million times faster still take as much long to boot as the machines that were a million times slower. <laughs> Come on, at least, you know, let's be ten times better, you know, if the underlying capability is a million times better, at least we should let through a factor of ten somewhere in improving the user experience. So the same thing, certainly, I didn't waste as much time coding every bit on the 6600 but by using the techniques, using the efficient parts of the old experience, and well it was a super job I thought.

**Booch:** Now Butler got you involved in the ARPA Project Genie around this time.

**Simonyi:** Well, Butler was pretty impressed and no, I wasn't involved in Genie, of course I met the people there but it was Berkeley Computer where we really started to work together. So when the professors started a company, I could join and again, they were appreciative of my difficulties with the INS and they were helping me in every way that they could, so that was really wonderful. And I learned a lot. So I was at Berkeley Computer until, you know, '70, '71, I was at BCC, I had stock in BCC and I remember I was neglecting my studies quite a bit. I mean I realized that my career is going to be dependent on what I do, not necessarily what my grades are, so I was kind of optimizing. I didn't want to be expelled from school because then I would really be deported, but just kind of optimizing it to get through. Berkeley had a wonderful notion of you could challenge a course without taking it, you could just challenge the finals, so I was taking fairly difficult courses, and I didn't get good grades on them. Those were the courses where I could learn something, and then I kept my grade point average at passing by challenging easy courses, you know, like all the computer science course, you know, I challenged a FORTRAN course or whatever, and it was no problem for me, being a professional, there was no problem passing those, and so I could afford to have quite a few bad grades. And in fact that dean called me once because in the first couple of quarters I didn't realize that you could game the system, so the first quarter I got straight As in really difficult courses in really difficult circumstances, but at that time, you know, I still had the money. So then it went way down, the dean calls me and says that, "Oh, Mr. Simonyi," this is in BCC times, it really went down, "Oh, you had such great results in the beginning and now your results are, you know, abominable, you know, what's going on? Is there anything that we can

do to help you? You know, for example if it's drugs or something, then, you know, we have programs and so on." He was very earnest, very concerned, so he asked, you know, "What is it, you know, is it drugs?" And I say, "No sir, it's not drugs, it's stock options." And <laughs> he said, "Well in that case, we can't help you." Yeah, and that was before the time when stock options were kind of well known or anything like that. 1:04:09

**Booch:** What was BCC trying to produce, and then they eventually did go belly up, didn't they?

**Simonyi:** Ay, ay, ay, oh, my gosh, oh gosh, but BCC was led by Mel Pirtle, a wild-eyed visionary and kind of a wonderful person but a little bit out there, and BCC was total hubris that it would be a time sharing system that would serve 500 users, that's why it was called the BCC 500, so it was Genie scaled up tremendously but with a terrific group of people. I mean that's where I met Chuck Thacker among others, and Chuck Thacker developed a wonderful word called biggerism, which is kind of like hubris but in a technical sense, and at BCC I learned a lot about biggerism and learned a lot about, you know, a lot of mistakes and worked extremely hard, extremely hard against pretty hopeless odds. So BCC of course collapsed.

**Booch:** Had you graduated by then?

**Simonyi:** No, not yet, not yet, I mean graduation was kind of slow. Yeah, I was going to graduate, it would come pretty soon, but no, BCC collapsed before I could graduate. But, hey, something good came out of it because that's the same time that Bob Taylor was organizing Xerox PARC. And of course Bob Taylor knew these guys. They were the apple in Bob's eye, certainly Butler and Chuck Thacker and then the many people at Berkeley Computer, very fortunately including myself-- who wouldn't have had any chance, being an undergraduate yet and having all the problems, visa problems to work at Xerox.

**Booch:** So what year would that have been?

**Simonyi:** In '72. So what happened was that when BCC collapsed, the main group went to PARC and Pirtle went to essentially do penance and he was given the ILLIAC project, the ILLIAC IV, otherwise known as the Vietnam of computer science.

**Booch:** Now why was it called that?

**Simonyi:** It was called because, again, I mean if the BCC 500 was suffering from biggerism, well the ILLIAC IV was biggerism times 64, you know, there's 64 processors.

**Booch:** That's amazing.

**Simonyi:** Yeah, well it was, you know, a bridge too far if you will. It was an amazing concoction of 64 processors that were implemented in ECL logic at the bleeding edge of technology. It was a really horrible design. Just to give you an example, the central processor had, you know, 24 bits and two's

complement, and the peripheral processors were 64 bits and one's complement. I mean just in the same computer that relied on this subtle interaction between the central, you know, it was the one instruction stream, multiple processor method of operation where the multiple processors were executing the same instruction stream that was issued by the central computer, the control computer, but they could disable themselves, so that's how you did conditionals. You know, it's kind of an interesting historical model for, you know, it's like one of those hopeless aircraft that never really flew, but it was interesting to try because other things were tried and they worked. But this one not only was kind of a difficult concept but it was implemented or executed in a very poor way. But hey, it was government, so I figured that if I worked for the government, then I won't get deported, which turned out to be incredibly helpful. I mean there were times when we pretty much had to call DOD before, you know, before I would be in deep trouble. So it was good for me to work on the ILLIAC.

What happened was so I was working half time in school, quarter time at the ILLIAC and quarter time at Xerox. Okay? That was the life, so it was quarter time for the protection and <laughs> it was necessity, what can I say. But I learned a lot. I did a fantastic project at the ILLIAC, which is pretty unprecedented and we wrote a program where we could process the wire lists of the design of the peripheral processor of the ILLIAC, create a simulator from it that simulated the working of the peripheral processor from the wire list, compile them and have the simulator simulate all the single point failure modes of all the components and then run the simulator on the ILLIAC itself in parallel operation. You know, you could simulate 4,096 processes in parallel on the parallel computer. Of course one of them was failing, so you excluded that, but it was still, whatever, it was 4,020 simulations you could run in parallel. And by that you could compare the actual observed results with the simulations and find out which single stuck-at-fault would explain the precise failure. So in that case we could go from a bad result of a test program directly to the chip and the type of failure that that chip had.

**Booch:** Amazing, amazing.

**Simonyi:** You could do that of course on discrete logic, where each component had maybe ten gates in it, so it had only like 20 or 30 single stuck type failure modes, which were the most common failure modes. But I mean just the, you know, that's the project where Hungarian naming was born.

**Booch:** Really?

**Simonyi:** That's right.

**Booch:** Now tell me what forces led you to Hungarian notations?

**Simonyi:** Well, that was very clear, I had to have this horribly complex program that I would have to write in PDP-10 machine code and it was clear to me also that the wire list wouldn't be correct. So I mean I never thought that by compiling a wire list, I would get a correct simulator. I would get an approximate simulator, which then would allow us to debug the wire list. So I had to write a debugger, and I knew it from the outset, so we wrote a very nice symbolic debugger and then I sat down with real engineers who had the drawings and for example I said one plus one and it came out, you know, three from the wire list, not without simulating any errors, just from the fact that the wire lists were wrong. Right? I had to find out



what's wrong with the database so I called the engineers and interactively he said, "Tell me what's this pin here?" I'd go click, click, click, I say it's high. "Okay, what's the pin here?" Okay, I say it's low, and we go on for awhile and he said, "Oh, I know, this resistor here is broken." Then I said, "Thank you very much" and I had to tell him that there are no resistors, nothing is broken, it's a simulation but I will find out why, you know, we found the place where the error was and so that's how we fixed the wire list. So it had to be a debugger and to write such a program in machine code is a substantial challenge, and you want to introduce a typing system, and how do you introduce types into machine code? And the answer is that you simply put the types on the names, and that's how Hungarian got started is that to use types in assembly code, in that particular case. I still have the listings of that compile. It's a substantial program.

And the other interesting thing is that because these are types that you create, you use a type system that is convenient to you, you are not stuck with a type system of an arbitrary language, so they become domain specific types. In particular, and if you think in terms of simulating a circuit board, a connector number, that would be a type. A pin number, a chip type, those were all types, so there was CT was the chip type, you know, DP was the dip identifier, you know, CN was the connector number and so on, so forth. And if we wanted to say which connector is connected to which pin or which chip, then you have an array of connector numbers that maps into a chip and so on, so forth. So it was an MP for map, CN, which is the domain, the connector number, DP or something like that. So the program kind of wrote itself, and the data structures wrote themselves by using those conventions. And then of course when I realized that this makes sense, not only in PDP-10 machine language but the next time I wrote another program in my other quarter time job, which happened to be Bravo, we used the same conventions, but, you know, nobody heard about them at the time.1:16:00

**Booch:** Because it was domain specific, did you find then that the engineers were able to grok what you were doing as well too then?

**Simonyi:** No, no, no, not at all, but certainly the engineers groked when I interacted with them with the symbolic debugger. At the time of the output, I told them the exact, I used the same conventions, so for example I don't quite remember if it was like D5 or, you know, C7 or something like that was the way that you identified the particular chip, but I read them out the exact domain specific names. The types themselves is a way to store the names, not the names themselves, so it was more a question of bookkeeping at the program level. But it certainly encouraged the direct relationship between the program structures and what made sense to the engineers, so I think that we have a valid point that indirectly it helped that relationship a lot.

**Booch:** We need to change the tape and we'll pick up then to your career at Xerox.

**Simonyi:** Xerox, yeah, hello.

**Booch:** Let's take a side step to your experience at Xerox, I mean you met some incredible people. There was Butler. There was Alan Kay, Bob Metcalfe around that time.

**Simonyi:** Well, Bob came a little bit later.

**Booch:** Okay.

**Simonyi:** Yeah.

**Booch:** So what were you asked to do when you first joined?

**Simonyi:** Well I joined very slowly as it were, beginning quarter time in '72. There was this wonderful tradition. I don't know how serious it was but it was 100 percent enforced that new people had to do a shit job and that's what it was called in the vernacular.

**Booch:** What was your shit job?

**Simonyi:** In the vernacular that's right. So my job was that Xerox was building this computer called the MAXC, M-A-X-C that you probably heard from others, Multiple Access Xerox Computer. It would be a very significant machine because the Alto is really used the same kind of hardware core so to speak. So the MAXC was going to be a PDP-10 and as you probably heard that has a story too. Of course, DEC was a competitor to Xerox because Xerox had Xerox Data Systems so it was politically completely incorrect for Xerox to have a competitor's machine for their research. But the researchers said they've got to have it and they were right of course. They had to have TENEX, the operating system that runs on the PDP-10. So they decided they were going to build their own PDP-10, the MAXC. My job was to bring up TENEX on MAXC, which was actually very similar to what we were doing at BCC. I was working with Chuck Thacker except it was much, much more simpler and Chuck did a wonderful job, cut down everything to size and really made it very reasonable. I remember him enthusing about this. You know these connectors, what's the name of that small connector that is on a phone, on the end of the phone plugs?

**Booch:** RJ something.

**Simonyi:** Yeah.

**Booch:** It was the RJ-11.

**Simonyi:** RJ-11 and also but there's the seven pin and then there's the five pin. At that time only the five pin was available. That was brand new and he was enthusing about it and he says, "We're going to do-- everything is going to be connected with these connectors and they never fail and they are so cheap and they can bonded" and so on and so forth. And, of course, that's when the first larger chips that were 64-bit appeared. There was a 64-bit register chip and then the 1103, the 1K dynamic MOS RAM. I mean that was an amazing idea. So things took shape and all I had to do is beat on everybody to make it exactly like the PDP-10. I mean my job was, of course, to adapt the TENEX to this machine and as long as the machine is the same as the PDP-10 the adaptation would be easier and easier to the extent that it's similar. So I was just running around making sure that everything was completely compatible which was kind of especially in the peripheral area was difficult to do because the programmers or engineers

that were working on the peripherals they wanted to improve things. And so I said, "No, no improvement." Everything worked. I think that Paul Hackel [ph?] was working on the same project. He was working on the peripherals and on a Nova so that was a Data General Nova that was running the peripherals. Since you don't want to interface all the peripherals with the main computer so what you do is you interface your machine with a computer, a mini computer, like the Nova, and then all the peripherals interface with the Nova and this way the hardware interfaces are minimized. However, with the software interfaces at least there is a degree of freedom there and you just have to insist not to exploit that degree of freedom and emulate it exactly. So we all did a good job and MAXC was running TENEX perfect and the corporation saved face. The group got some confidence especially after the BCC-500 that took a long time and wasn't successful. This one took a very short time and was very successful. Of course it was much simpler to be sure but on the whole it did almost the same job so, yeah, the technology kind of caught up a little bit. The best technology was used so it was a great project making MAXC work. And, of course, that's about the time when Bob Metcalfe joined.

**Booch:** Was Alan Kay around the same time?

**Simonyi:** Oh, yes, Alan was absolutely there and I mean it was a delightful environment. Bob Taylor was playing touch football with us all the time and I'm eternally grateful among all of his other contributions, he basically started the Internet. He wrote the first check for ARPANET let's put it that way and so he certainly has his role in that development when he was at ARPA. And then at PARC he was, of course, responsible for the personal computer, at least the first personal computer, so two of the most significant things of the 20th Century he had kind of a direct input too and it's amazing. The third thing I'm very grateful for is teaching us, or teaching me at least, the rules of football and playing touch football in a very, let's see, intelligent way and you appreciate the intellectual content of football rather than the physical content that, of course, we weren't very adept at.

**Booch:** What do you remember about him as a person? He had these incredible accomplishments. What kind of a man was he?

**Simonyi:** I think that he was a very intuitive man. He was majoring in psychology. He reminded me very much of Neal Silverback, the director of RegneCentralen. I call these people, people collectors. They are the nodes in this scale-free network of talent and people in the world. So what Bob brought to the table is kind of this intuitive vision of the future. Butler can talk about that very well. He wrote once that he knew that when Bob said something about the future he had absolutely no logic or reason to say what he was saying but he was right. So he was very great at intuiting the future and then, of course, he was a great people collector. I knew three world class people collectors in my life which was Neal Silverback and Bob Taylor and the third one is Bill Gates.1:26:20 - one hour to do 10min of tape

**Booch:** And we'll talk about him in a bit.

**Simonyi:** Yeah.

**Booch:** You mentioned statements about the future and Alan said some things that you disbelieved at first about the computer.

**Simonyi:** Right, that's absolutely true that Alan had a clear vision of the Dynabook and he was talking about it all the time. This was when the Alto was already thought about somewhat. But he was way ahead of that and he liked these extreme statements. One of them he was saying that we should think of computers like we think of Kleenex. You use them once and then throw it away. Now that's clearly an overstatement but he was doing that to make people feel more relaxed about computers. But the other one that wasn't that extreme but still I didn't believe was this idea of the Dynabook. And he said that one day there will be a device that is about as big as a book, weighs as much as a book, and you could read it like a book and it would have the power of a PDP-10. Of course, the PDP-10 was a good measure at the time because MAXC that we just worked on, which was as big as a sizeable cabinet and that was the standard measure for computational capability. Well it turns out that, of course, this device has the power of several hundreds, if not maybe ten thousand, I don't know. Actually I'm getting confused about the orders of magnitude but we are talking about between 100 and 100,000 times the size of a PDP-10. And what really bothered me about that prediction was not as much the computing power but the display, the fact that you would do a display that is flat that would somehow wouldn't crush your leg was pretty far-fetched in my humble opinion. But the idea, to me the idea of a personal computer wasn't strange at all. To me a computer was always personal. Being a systems programmer and, of course, the Ural where it was all mine, completely mine. I mean I cleared the whole memory when I started my session. But this business of having a flat screen and that's amazing I think. Well it's amazing that he's right and then it's even more amazing to the extent that he was under estimating the progress, as opposed to over estimating, and it's also amazing that even the under estimation seemed so improbable to people who were like let's say, if I may include myself, who were kind of at the edge of development and could imagine a lot of things.

**Booch:** Now around this time you began Bravo. Tell me how that was first conceived.

**Simonyi:** Bravo was an amazing project when the Alto appeared. Of course everybody remembers different, successful things. But when the Alto appeared there were really two ways of doing the office of the future. There was the two different labs of Xerox, the SSL and CSL, computer science lab and system science lab. The Alto was more a research vehicle that was very much supported by the Smalltalk people in SSL but it wasn't thought of at least initially as something that would be for the end user. People thought that a timesharing system of say a shared system of Nova computers that was code named Polis (sp?) was the center of development for the office. So even to suggest that Bravo should-- I'm sorry Alto should have an editor on it was somewhat, you know, it implied duplication of effort. There was such an effort already going on in Polis (sp?) so it wasn't a good idea to propose that. But the Alto needed an editor and it was clear that quite a beautiful editor could be written for the Alto and remember the Alto had a black and white bitmap display. It had a mouse. It had an un-decoded keyboard. It had all the ingredients that are necessary for WYSIWYG. And, of course, then the network came around. The laser printer was coming alive. So all the components were there except for the software. Butler Lampson had some notes of how such an editor should work and he listed a couple of key ideas, one for how to update the display efficiently and a second one is how to edit the documents efficiently using a technique that came from Jay Moore's work, who did an editor in Lisp and to see, for example, if the methods of artificial intelligence would be helpful to users of editors. It wasn't WYSIWYG at all but it had this technique of so-called piece table (?) that was a very efficient technique which you needed if you were programming in Lisp and wanted to create a useful system. That's also kind of interesting that by doing something under restricted circumstances you create innovation that if you transplant in a richer environment but maintain the efficiency then you can get tremendous results. It's when you use the richer environment to relax and use sloppier methods then you get kind of sloppy results. Usually you get a little bit better results because it is a better environment after all, but we're not going to proportional to

how much better it is which is a great tragedy of computers, of course, and which is still continuing but it means that there are still opportunities. So anyway, there was a tremendous amount of tension of who would implement it, how would this work, and so on. And I think I came up with the idea that since I had to do a Ph.D. thesis at the time and Xerox was happy to do something that had to do with the Xerox or politically we couldn't say that we're going to do an editor for Alto but we could say that we're going to do a Ph.D. thesis. There was no controversy about that. So and, of course, that's the name Bravo. It was the second experiment in my Ph.D. thesis.

**Booch:** What were the first experiments?

**Simonyi:** It was called Alpha of course.

**Booch:** Of course. Of course.

**Simonyi:** Thank you for clarifying that.

**Booch:** Just to make sure there's no confusion.

**Simonyi:** There is no confusion. So that's how it came about so we hired a couple of summer students and one of them is Tom Malloy, who is now chief of chiefs at Adobe. And by the way, talking about-- so I mean Adobe of course came from Parc so it's wonderful, wonderful people. And we set out to write an editor and we finished it about three months, the first version that didn't have formatting yet but it was attractive enough that Larry Tesler kind of picked up on it and stopped his work on Polis (sp?) and used kind of the framework of Bravo to implement Gypsy. Gypsy kind of referred to the fact that it was called Gypsy Bravo that it left the home of Bravo and a separate code base was created from it. And so he developed bold and italic and just the font synthesis that worked really well. And then in '75, early '75, we came up with the Bravo 3 that reintegrated the formatting capabilities in a highly efficient way into the framework and that's when Bravo, whatever 3.1 was kind of the first really useful version that had full formatting and was connected to the laser printer and it really worked like a modern word processor. About that time we had a lot of visitors, started to get visitors from important clients that were shown the research lab and we had a standard demo where we printed. We had a memo that was on the screen. It has the Xerox logo on it. Of course, the Xerox logo was still text except in the specific font called the logo font. And so the memo looked terrific on the screen, black and white though, and we printed it on the printer and we printed it on transparent stock so we had one of those memos on transparent stock pre-printed. So when we gave the demo and we edited the memo and so on and so forth, at one point we pulled out the transparent, pre-printed transparent stock memo and put it over the display.

**Booch:** That's showmanship.

**Simonyi:** And that was fantastic and that's when one of the visitors said, "I see it's what you see is what you get" which was the tag line from Flip Wilson that's right from Laugh In. And that's where the phrase WYSIWYG started at one of those demos. It wasn't our phrase but it was certainly this demo that provoked that. An interesting part of the demo is that the printout and the screen weren't alike but they

were alike enough and I will tell you in a second why not, but they were alike enough so that it was very impressive and certainly the logo was at the right place and the overall effect, the margins, the logo, the justification, all that was the same. The details weren't the same because the printer, of course, had much higher resolution that we wanted to exploit and the screen had a fairly modest resolution by today's standards but still very, very impressive for that time and especially the fact that it was black and white rather than green on black, for example, which was kind of more the standard. But still with that difference, if you will, people didn't notice it at all which we thought they wouldn't and, of course, that's not what's important. The important thing is just when you format a document you are looking at an overall impact. You are not looking at the exact line breaks.

**Booch:** What was Bravo written in what language?

**Simonyi:** Bravo was written in BCPL, which is a fantastic language. BCPL, of course, is a predecessor of C. After BCPL came B and then came C and nobody can figure out if there had been a successor to C would it be D or would it be P.

**Booch:** And by the way, I see Microsoft created a new language called D that just came out.

**Simonyi:** Oh.

**Booch:** For declarative.

**Simonyi:** Oh, declarative.

**Booch:** Yeah, yeah.

**Simonyi:** Oh that might be. That's because of B and C, whatever. BCPL was a type less language. There was a single type which was a 16-bit word. It was compiled for the machine code of a Nova and, of course, the Alto was running, emulating the Nova instruction set, not that it matters.

**Booch:** Unless you continue the Hungarian notation.

**Simonyi:** Absolutely.

**Booch:** Because you had a type less language and refined it further.

**Simonyi:** Exactly, so well, of course. Then we went over to the typed languages too later on because the typed languages didn't give you the types that you needed, the intentional types or the domain types weren't in the type languages, so it was still important. But I remember if you showed me a Bravo listing, I would immediately, we would look at one name and I would tell you exactly a lot about that quantity. I

mean I remember CP for example was the character pointer. One of the key types, PC was one piece in a piece table and so on, so I'm sure there was some map from CPs to PCs.

**Booch:** So simultaneously you--

**Simonyi:** But I mean it's amazing with a couple of rules. I mean I may not remember all the Hungarian tags but show me a listing, we immediately recognize it because there are about six or seven key domain types in Bravo and then all the names were a combination of those types.

**Booch:** So simultaneously you finished up your doctoral thesis.

**Simonyi:** Yes.

**Booch:** And that was on meta programming so tell me kind of what your focus was there.

**Simonyi:** Well, my thesis wasn't-- I don't think it's a very good thesis and I don't think people should read it except, for example, for the Hungarian part. I quote that all the time and it is still very, very useful and now I think with intentional programming we see that it has important theoretical background. But meta programming was just a way to-- well it was expressing my interest in the productivity in trying to make the programming process more regular maybe and the idea was that you create a spec that is more detailed than what we called specs before. It's almost like a program. Call it maybe pseudo code. And the purpose of that was to establish a closer relationship between people who worked on that code. Of course I was suggesting a hierarchical organization because at that time people were talking in terms of chief programmers. But if you ignore, I can explain it in today's terms that what I should have been advocating is, for example, pair programming that is agile programming. It's an important concept so I was getting a little bit closer to the idea of on one side to the intentional input and process description of programming and this meta program would be kind of an equivalent of what today or not equivalent to but should have been in today's terms the intentional input and the working relationship should be more in the line of the pair programming rather than a chief programmer and a coder if you will.

**Booch:** Indeed using the chief programmer term that's Fred Brooks' term.

**Simonyi:** Yes.

**Booch:** So you were starting to have a contrarian notion to Fred's experience to some degree.

**Simonyi:** No, no not at all. I was following at that time.

**Booch:** Okay, following, okay.1:45:26 - 70min to do 19min of tape

**Simonyi:** Yeah, following yeah. No, no it's explicitly I was saying that it was explicitly influenced by that and I was explaining why I disagree with myself and I disagree with Fred. But at that time I was agreeing with or at least following that one. Frankly I didn't pay as much attention to the thesis as maybe I should have because I was focusing on the price, if you will and to some extent we did use the thesis to do Bravo. So my best energies were oriented, focused on Bravo, not on the thesis and I think that the quality difference is substantial.

**Booch:** So as the '70s unfolded give me an update on your immigration status.

**Simonyi:** Thank you very much. So what happened was that working on the Illiac paid off and slowly I got the green card and in 1975, January, I got the green card and the next day I quit the ILLIAC project and I think that was fair. I mean I think it was wonderful that they helped me even though they knew that I might leave but the ILLIAC project was coming apart anyway. They really didn't need my continued suffering, co-suffering shall we say, and in some sense it was the payment. Their helping me get the green card was payment for my efforts that were and I did do a good job. I certainly put out so I think it was fair to everybody involved but I did quit the day after I got my green card.

**Booch:** So there was a sense of relief then because finally you were straight with immigration.

**Simonyi:** Oh, my gosh, green card was the best thing, yeah. And then, of course, the countdown to citizenship starts and it's a five-year minimum. It turns out that I changed jurisdictions meanwhile by moving to the state of Washington, which we'll discuss so finally I got to be a citizen in '82.

**Booch:** So before we leave Xerox, I mean things were winding down for you in the '70s, do you have a Bob Metcalfe story because you spent some interactions with him I'd imagine too? What kind of a fellow was he?

**Simonyi:** Well, Bob was and is a wonderful guy. There are many Bob Metcalfe stories. I mean he was driving a Morgan, have you heard that?

**Booch:** No, I haven't heard this one.

**Simonyi:** Oh, the Morgan is this wonderful British car that has the wooden under carriage. It's a sports car with a wooden frame to which the rest is attached by leather belts and, of course, this was very difficult to import but he got it driven. It was converted to propane and so it could be imported. It could pass all the tests running on propane so he was always looking for some place to get propane for his Morgan. He was incredible. I'm not going to talk about Ethernet he did wonderful things but I'm sure he'll discuss it.

**Booch:** He'll tell that story.



**Simonyi:** The great stories of the name and the idea and the improvements and what came from where. But what was I going to say? Oh, yeah that he was really venturesome and he quit Xerox and went into banking and became a banker.

**Booch:** Didn't know that.

**Simonyi:** That's right and he became a vice president at one of the banks because the bank decided that communications would be very important in the future, rightly so. I'm sure they were working on early ATMs or so on. But he acquired the trappings of a banker. I remember he then came back to Xerox with a greatly improved resume and I assume salary. And he described to me his past, for example his office where he had a button under his desk where he could close the door so the door was always open. But if I may tell you this in perfect confidence he would press the button and the door would shut and presumably the visitor would be incredibly impressed about the import of the information that is about to be conveyed so that was-- I remember that.

**Booch:** That's marvelous.

**Simonyi:** He had a great laugh telling it and, of course, we were envying him for his worldliness of experiencing these exotic things like being a vice president at a bank.

**Booch:** Marvelous.

**Simonyi:** It was way beyond our horizons.

**Booch:** Now you yourself decided around this time to move on from Xerox. Can you tell me the decision process and what led you to move up to this small company?

**Simonyi:** Sure. Yeah, it will connect with Bob Metcalfe, of course. The decision process it started from frustration. Of course, I was horrible in selling my ideas. I don't know if I should say this but well I'm not going to talk about it. In '75 when things were going really great, I wrote some memos about the future and these ideas, at least these memos, became fairly handy later on. But I think that they had most of the ideas-- just ideas that I've heard or just a collection of the ideas of what possibilities does this open, the Alto, the so called personal computer. I don't know what we called it at that time but they call it a personal computer. What kind of programs you expect to see on it and what we can do to make this happen? It's a little bit like my first day at RegneCentralen where everybody was scared of what we're going to do in one month. And I simply asked do we have input/output conversion routines? And the answer was no. Okay then let's write it today, at least get something done.

**Booch:** Do you remember like two or three of those ideas that you felt--

**Simonyi:** Yeah, I think that something like the spreadsheet. It wasn't called a spreadsheet but it was called spiders. I've heard it from Peter Deutsch is that things dependent on each other that you evaluate

something and you put in a lot of stuff that depends on each other and it evaluates them but just in terms of the combining of putting graphics into documents and combining the results of calculations with the graphics just having footnotes, for example, I mean having maps.

**Booch:** Would it be fair to say politely that those ideas fell on deaf ears in Xerox?

**Simonyi:** Oh, absolutely, yeah. I mean for several reasons. One of them is that I was again absolutely not a good communicator. I think that my writing was pretty clear but it was more concentrating on the results than on the arguments. I don't think it was presented well. I think that we would have to have an equal vision. An equal visionary would get excited about it, but somebody who is a manager and evaluator would not react to it. So this is not a criticism.

**Booch:** It's a reality.

**Simonyi:** In some sense well if I wanted to criticize I would say that well it's their job to look at bad communications as well, especially in an area where there is clearly something is brewing. I mean the results were it's not by accident that they took the clients, be it Citibank or for that matter we had installations of Alto and Bravo X in the government. We had Altos at the executive office of the president. We had one in Congress right next to the dome of the capitol. There's a little room that I visited. There was a laser printer there, a Dover, and an Alto.

**Booch:** Who was the first president that would have been near one of these?

**Simonyi:** It would have been Carter but I don't think that these were the Carter years, of course. The story I've heard though that when Reagan came into office, of course, there was a big housecleaning and this thing was going to go but then somebody showed a printout of a speech in a very large font and, of course, Reagan having poor eyesight, you know difficulty reading just like I do, being able to see it in a large font was an incredible benefit so that for him the speech could be printed immediately. Anyway, according to this story it was kept for a while.

**Booch:** Marvelous.

**Simonyi:** But visiting, that was a great one and a half year.

**Booch:** ...tell us a story of a laser printer or a laser device.

**Simonyi:** Well, this was the printer that was installed in the capitol building next to the dome and we had people from the government printing office, GPO, that were trying to use it, and in particular see if they couldn't use it to produce the Congressional Record, which is a considerable effort, and I'm talking about a person, a printer person with ink-stained hands. This was all done in hot metal, the Congressional Record, and he proudly showed me a page that was printed on the laser printer and prepared on Bravo that was a page from the Congressional Record, you know, gorgeous, gorgeous, you know, it has the

gothic fonts and all the traditions exactly like the legacy dictates, which it has to be. You know, it produces every day, I mean it's a giant operation. So I look at that page, and it's a gorgeous page, and I notice in it features that are on my list, you know, the footnote or the subtle layout features that are typical for high quality books that Bravo wasn't able to do, and it was on my wish list, of course, all of them, everything was on the wish list, all the typographical, fancy stuff, equations, you know. So I'm taken aback, what's going on, and I express my puzzlement to this printer, I say, "I don't know what you are trying to pull on me, am I on Candid Camera?, you know, Bravo cannot do this." And then he looks at me like strangely too and he thinks that his leg is being pulled and, "No, no, I've just done it, you know, yeah, I've done it on your system." And so, you know, I start to think that he's not crazy, I'm not crazy, there must be some explanation here, and then I look at the system and of course the screen, it looks horrible, the document looks horrible, and I say, "Oh yes, we have this special debugging feature where we can give absolute positions to certain text elements, which are enforced, which are not WYSIWYG but enforced only on printout, and he was using these kludges, if you will, to assemble his document completely by hand. I mean he didn't have any of the features, well, he had some but it wasn't invariant if you will. If you started to edit it, it would all go to hell, all the page layout would go to hell. And I said, "Oh, oh well, you were cheating, that's not very interesting," says I, and then he looks at me again crazy, "You are kidding? You should see what we are doing today. This is much better." So this was a big lesson to me. Of course doing it by hand on a computer may not be, you know, the perfection that computers are capable of, but it's much better than doing it by hand in hot metal. So it was a big improvement step for him, and I completely ignored that intermediate possibility in my wish to have the perfection, and I didn't even recognize that it would be exploited, and it was just a big lesson to me that less than perfect is still valuable, and in fact perfection sometimes, it delays things so much that meanwhile people keep suffering under the kind of the old methods.

**Booch:** So here you are, you're about 33, Bill was 26 and you made the leap to go to Seattle.

**Simonyi:** Well, okay, so what happened was that, you know, Bravo actually didn't go anywhere, well of course Star was being developed at the time, I was working at Star basically out of necessity, I was very frustrated, I saw biggerism again and that was, you know what, it's great to have failed once or twice, you know, seeing it at Berkeley Computer, maybe seeing it at ILLIAC, not necessarily from my contribution but from others' contributions, all the lessons from Chuck Thacker, they were all on the wayside. I think that Star was biggered to an unbelievable extent and I couldn't contribute. And I saw that this is not going to work, this is not going to work. This time I'm not worried about deportation, I have my green card, I have to make up my own mind, I have to play a different game, because this game is rigged, we are not going to win. So I went to Bob Metcalfe. Well actually I did realize that there was this personal computer thing, and in fact there was an Atari...no one with one of those Chiclet keyboards, which one was that, the Commodore? I think the Commodore.

**Booch:** Yeah, the Commodore, the Commodore PET.

**Simonyi:** Yeah, the PET, Commodore PET, that's right. It was, you know, the SSL people got one of those and, you know, Alan was probably showing it, Larry Tesler was showing it and we were chuckling about it, you know, how primitive it was and, you know, it's silly and our stuff is so much better, and then one night I met Heckel, Paul Heckel, who was working with me at Berkeley Computer and on MAXC and he said, "Come to my place and check out this thing, VisiCalc."

**Booch:** Dan Bricklin's work, yeah.

**Simonyi:** And I went and he showed me VisiCalc running on the Apple II, and you know, my jaws dropped that wow, here is this primitive machine and he's doing something very well that we were just dreaming of, my memo was kind of hinting at, but it's not even in the..., there's no way I can get resources. This is big, and imagine if you can do this with this machine, what could you do on a more serious machine like the Alto or the machines that are coming. So, I said I've got to get involved with the personal computer arena and I went to a seminar that was held by Vern Rayburn, who was a VP at Microsoft and I just listened. He was talking about just the ecology that is building up around personal computers, and I could see A) that it was exciting, B) that they didn't know about the real good stuff that I knew about, and again, I was thinking of combining, you know, their dynamism and my knowledge, if you will, sorry, the knowledge in my head, it's not my knowledge but just the knowledge in my head, interesting things could be done. So I went to Bob Metcalfe, the guy who has been outside <laughs> of the reservation before to ask for his advice, and we had a lunch at Stanford and I didn't want to talk to him at work, so we went outside of work and said, "Well, you know, time for me to go to a bank or something. What's your advice?" He said, "Oh well, you know," and we made a list and, you know, he put down Microsoft Bill Gates, crazy guy and then Apple and personal computer, which was VisiCorp, become VisiCorp and, you know, a couple more companies, you know, the CPM company, what was it? I can't remember. Digital Research?

**Booch:** I know who you mean, yes.2:07:20

**Simonyi:** Yeah, and so he had it all down. Of course I could have gotten the list from a magazine. Maybe there was no magazine at the time, but there was Byte.

**Booch:** Byte was around, Popular Electronics was around, too.

**Simonyi:** Popular Electronics, but....

**Booch:** They didn't know anything about software.

**Simonyi:** So anyway, but Bill was the first on the list and next time we had an installation at Boeing.

**Booch:** Up in Seattle, yeah.

**Simonyi:** And so I went to upgrade Boeing or whatever with the latest, or fix some of the problems they had or observe some problems that they had with Bravo X and I dropped in at Microsoft.

**Booch:** Where was their office around then, do you remember?

**Simonyi:** Of course I do. <laughs> It was an 8th Street in Bellevue in a high rise at the 8th floor, one half of one floor in this high rise building that is still there. Now it's kind of the smallest building, at that time it was the biggest building, and in fact I was driving by the building about three or four times, I was looking for an address, and I didn't bother to look at the high rise, I was looking for kind of a more a lower end building that was all around that area. And finally I went to the bank at the bottom floor asking maybe can they help with the address, and they said, "Oh, it's right here."

**Booch:** You just dropped in unannounced?

**Simonyi:** Okay, well, I assumed that Bob called, but yes I was unannounced, yes. I was under the impression that Bob called his buddy Bill. <laughs> And it was unannounced, and but I had, you know, a little demo package, which were all the published articles about the Xerox systems. You know, there was one in the Seebold Report and so there was a little bit of publication about it. And I got to talk to Steve Ballmer right away, you know, you just, you know, but at that time, you know, Microsoft was hungry for talent, and I know because when I joined, I mean we spent most of our time looking for people. So yeah, I mean I saw Ballmer immediately and great guy, you know, super energy, and then he says, "Oh, you've got to talk to Bill but Bill is with this Japanese OEM's, blah, blah, and then but meanwhile talk to Paul" and then so I talked to Paul and then I get to see Bill, but it's time for me to go to the airport, the plane is leaving, so fine, Bill is going to drive me to the airport, so Bill drove me to the airport and, you know, he said, you know, "BCPL, C, we're going to do this, we're going to do that," the guy just, you know, saw it all and he had a plan of doing something about applications. At that time he only knew that what are you going to do about operating systems, which we will see in a few minutes, but applications, he knew that it needs to get done but didn't know how.

**Booch:** Do you remember the date of this meeting?

**Simonyi:** Yes, it's 1980, September, October. I don't know the exact date like with the July 17th, but yeah. It was kind of an impromptu meeting, and then we had two other meetings with him and in one of those meetings I came up with a business plan for Microsoft, which I still have, and that one has Word and Excel, not with those words but literally it's there with graphical user interface, it means that we are going to be a world-wide supplier of graphical user interface software, application software. So all right, and at some point in this time, and I'm not going to say exactly when, but Bill said, "Have you seen the chess machine?" Have you seen the chess machine, like it's obvious.

**Booch:** Mm-hmm.

**Simonyi:** And this was a time when chess was one of those emerging, you know, application technologies. I don't know if you can remember.

**Booch:** It wasn't a solved problem, there was Sargon.

**Simonyi:** Yeah, Sargon, all of those things, yeah, but it was a popular product if you will, chess, and so I said, "Bill, I'm disappointed, you know, I'm not interested in chess, that's a game, that's a fad, there's no

money in it. It's applications, applications, you know, word processing, blah, blah, blah." Bill looks at me like, you know, who is this guy, what the hell is he talking about? He says, "Charles, you don't understand, you don't understand nothing." So he goes to a closet and opens up the closet. Okay? And inside the closet is the IBM PC, code named Chess. <laughs> You know, it's like somebody opening the Hope Diamond, you know, have you seen a big chunk of coal and I said, "Bill, we aren't interested in coal, we are interested in more important things," and then he opens up the cabinet and there's the Hope Diamond. Okay, <laughs> excuse me. No, I have not seen that chess machine. So that was the, I don't remember if you remember the first PC, it actually had a very nice keyboard, it had a keyboard that was like.

**Booch:** I loved the feel of it, yeah.

**Simonyi:** Yeah, the feel of it but also the shape of it and it wasn't plastic, you know, it was kind of that whatever light metal cast blah, blah, I mean you could see immediately that it was IBM and the quality and you know, it's a step forward, you know, that was the acceptance of the personal computer by, I mean that was like the biggest secret in the world. Obviously the code name, I had never heard the code name before. There were two, you know, there was Go and Chess. Go was done in Japan and Chess was done in Florida, and Microsoft had one of the Chess machines in this closet and they actually had to put chicken wire on the ceiling so that, you know, it couldn't be, you couldn't easily climb into it or spy into from the false ceiling. <laughs> So, you know, they had unannounced security visits from, I mean IBM had the contract, included that they have the right to have security checks unannounced, and they did have, you know, every once in awhile somebody came and said, you know, "We are from IBM, here's the contract, we want to check your security." So that was that. So this was an incredible period of time.

Of course I had to wind down, so we basically had developed an agreement, I got my offer in a FedEx package on Christmas Day in 1980, or actually got it on Christmas Eve but we weren't home so we got to drive in on Christmas Day and pick it up. FedEx of course was a brand new thing at the time, I mean just this whole notion of having overnight something <laughs> was an amazing idea. It was an amazing transitional time, and one thing that was incredible that while I was talking to Bill, of course I didn't mention, we were still negotiating and just clarifying things, so I didn't say anything at Xerox, and my project was pretty much finished, and Jerry Elkind, my boss, very nice man, he asked me to interview at Stanford at the corporate offices because there was a staff job that was available, and I said, "Jerry, I'm really not interested but thank you very much for this opportunity, but I'm not interested." I had to say that of course, but Jerry said, "You don't understand, you have to do this as a favor to me, because, you know, so do it as a favor for me." Oh, as a favor, sure, you know, so now I had a chance of why not. Okay, go to Stanford and interview for the staff job, and this was important because I was, almost simultaneously I was seeing these two different <laughs> prospects if you will and what a contrast, you said it, contrast, you know, one of them looked like death to me, I described it like, you know, the cancer ward, you know, where the people, it's the empty halls, you know, the long hallways, completely sterile, the questions were meaningless, you know, the vision was lacking. I mean it was total darkness if you will, not any disrespect, I mean I have great respect for the past achievement of the company and the intelligence of the leadership, but just from this specific, in a domain specific way, there was zero vision and obviously, I mean, that's why they wanted maybe a staff person but why fight uphill when at the same time I go to this other place where there's a pull, there's tremendous energy, there are tremendous resources, which is amazing. You know, well we are talking about hundreds of thousands of dollars or maybe even a million dollars, okay? I mean and Microsoft had royalty checks coming in for Microsoft Basic, you know, \$10,000, \$20,000, they were left unopened on the table because there was no one to

process them. Okay? There were more important things to do. Greg Watchin [ph?] was kind of the first guy who put that into an order was interviewing at the same time as I was and he was telling me this story that there are these checks unopened. <laughs> But that means that of course Xerox had hundreds of millions of dollars for development of new laser printers. Some of them were incredible technological achievements, you know, there was this machine that was doing like one page a second or, you know, an incredible high rate, but there was no hundred thousand dollars for software or, you know, to have a team to do what needed to be done.

**Booch:** What a contrast.

**Simonyi:** It's an incredible contrast.

**Booch:** You mentioned Visual Basic, did you run across Al Cooper at all, did you interact with him at all?

**Simonyi:** Much later, and I'm sorry, I misspoke, and thank you for pointing out it wasn't Visual Basic, it was Microsoft Basic at the time, yeah, Visual Basic came way, way, way later and that was my mistake. It was GW Basic.

**Booch:** Got it. Now also on the list that Bob gave you as Apple, did you ever reach out to Steve or Woz or were they off your radar?

**Simonyi:** Certainly I didn't do anything after Bill, that was it.

**Booch:** But they weren't part of the decision process?

**Simonyi:** I think that the key thing was that with Microsoft it was a software company, that was my attraction. You know, Apple was a hardware company and I thought that the idea of being at a software company where we are in the middle of it, we are driving it, that was an incredible new notion, and I very much wanted to be a part of it. Now I did meet David Marquardt, who joined the board of Microsoft and again he was making an investment in Microsoft as a venture capitalist and so we were kind of together at some time and together we went and listened to Steve talk and he was giving a speech about, this is 1980, he was giving a speech about fractional horsepower motors, very, very impressive. You know how before fractional horsepower electric motor, how it revolutionized manufacturing for example. Before that, you had to have kind of a large power source that was distributed by shafting and belts in a factory and it was the fractional horsepower motor that could distribute that power to many places and, you know, what it created and so on, so forth, and of course, you know, you can see, I mean it's such a metaphor today, of course, but at that time you had to use these kind of very obvious or striking, not obvious but then striking metaphors to communicate the idea that computing power needs to be in smaller chunks and distributed to where it's used instead of having it in a central. You know, you had to make those arguments and today nobody's making that argument because actually now we are doing the opposite because now we have server farms, but that's neither here nor there. But he was a very impressive character. Of course Larry Tesler was already at Apple and Tom Malloy after I left went to Apple, and they did great stuff. You know, interestingly enough, they worked on Lisa, not on Mac. Now we were, the

only time I worked with Steve was at the Mac days, which is a little appreciated fact that Microsoft was very much a supporter of Mac way before, just like we had the first Chess machine, we had the first Mac also outside of Apple, and we kept it in a closed room, which was code named Beach, because the code name for the machine was Sand.

**Booch:** Very good. So you then moved over in '81 up to Seattle.

**Simonyi:** That's right, '81 February I drove my car and started to build a team and of course started to work on Multiplan right away, which was, unfortunately there was already some Legacy code that we should have just thrown away, but it kind of set us back a little bit.

**Booch:** And Multiplan therefore was the first, the predecessor to the spreadsheet?

**Simonyi:** The predecessor to Excel, yeah, that's right. That was our first attempt at spreadsheet, where we were competing against VisiCalc. So yeah, I mean those days of Microsoft were incredible. We put together, we spent a lot of time hiring. I remember, for example, Bill was in the middle of everything, and we were hiring people so we were interviewing side by side <laughs> at Berkeley, for example, and Princeton. And it's funny that sometimes Bill was overhearing my questions and vice versa, and at that time I was very keen on this question of base negative two. In my earlier life I saw a machine that was operating in base negative two that was a Polish serial computer. Maybe the Museum could have some record of it, but it turns out that in base negative two, there's no issue about how to represent negative numbers, so you don't have to decide as it were, you know, two's complement, one's complement, sign magnitude and all of those things that were really occupying a great deal of bandwidth during the 50s. So these guys, being mathematicians, they said well, there's one way out of it, we use base negative two, where there's no sign and negative number representation is completely well defined. But I made an exam question out of it and just said, "How do you tell, I'll give you a binary number, it's in base negative two, how do you tell whether the number is positive or negative?" And of course the answer is that you count the position of the leading bit and if it's even then it's negative, and if it's odd it's positive or the other way around, but so because the interview of positions they are worth one, minus two, four, minus eight and so on, so forth. And you know, it's something that can be solved in two or three minutes and it really distinguishes people who really can follow up on some rules in a conceptual way and those that are completely frightened by something that they haven't heard before.

**Booch:** Deer in the headlight. Yeah.

**Simonyi:** And Bill <laughs> overheard this and says, "Oh well, that's interesting. Oh, we're going to get people who know this. That's great. I should ask the same thing." <laughs>

**Booch:** So Multiplan then finally was released when? That had to have been a big event.

**Simonyi:** Well, Multiplan we had in '80 September, we had a Multiplan on a Datapoint terminal, so we had the Multiplan coming up pretty quick. It's interesting, we had this dream at the time that we would create an interpreter. The way it looked at the time, there was this incredible variety of machine



architectures and there were 68000 machines, the 8080, there was a 6502, you know, there was different operating systems, there was a third one, the Z80, there was a Z80 and of course a different operating system for each and a different environment and somehow with the application it's fine for the operating system to work for IBM, but for the applications they should work on all of these. And because of this fragmented market, we thought we would attack it with an interpreter and then we would write stuff in C and interpret it and everything would be hunky-dory. And in fact it was to some extent. Remember what was it, Texas Instrument had that fancy machine TI99.

**Booch:** Yeah, that's right, that was an amazing machine.

**Simonyi:** Well, amazing machine completely different from anybody else. <snaps fingers> Multiplan came up right away on it with practically zero cost for us because all we did was we shared the interpreter specs with the manufacturer, they wrote the interpreter with their engineers. I mean, you know, it's a completely proprietary chip, Texas Instrument chip, unlike everything else, very complicated address space, but Multiplan worked and the checks were, you know, small checks, <laughs> but they were coming. So Multiplan was probably the most ported application ever. Now meanwhile, the IBM PC had become a big success and it started to have more memory too. Oh, the other thing about interpreted stuff that the memory usage was extremely low, because the interpreted code was super optimized, not optimized in a conventional sense, but just in the information theoretical sense, so the most common instructions were very small. That's a trick that we used, I mean we used as early as in the process control system in Denmark, but it was also used by Mesa [ph?] to some extent, the Xerox language, which had many problems, from my view it was suffering of biggerism, but in this particular way of optimizing the density of the code, it did a pretty good job. Well we did a much better job and in a much simpler, again we simplified the environment, we used the best techniques in a simpler case, then it gets incredibly good, and so our memory usage was very much optimized. The speed was of course a problem. We thought that we would be able to do something about it, but it was necessary. Yeah, yeah, yeah, so meanwhile the PC had become very popular and then it got a fair amount of memory and then this program called Lotus 1-2-3 was written by Lotus, by Ray Ozzie. Well, Ray was.

**Gardner Hendrie:** It was Jonathan Sachs.

**Booch:** That's right, it was Ray later.

**Simonyi:** Jonathan Sachs, yeah, that's right, yeah, yeah, he was using, not Opera but Symphony. Opera. <laughs>

**Booch:** Symphony, one of those musical metaphors.

**Simonyi:** Symphony, right. But yeah Jonathan Sachs wrote it in machine code and it went like a bat out of hell and it was awesome. It had a macro language, because now that, you know, you have speed, you can afford to put one more level, I mean we could not afford to put two levels of interpretation on something, so they put a macro language on it. I mean it was, it wiped us out.

**Booch:** This is around the time of WordPerfect, too, wasn't it?

**Simonyi:** Yeah.

**Booch:** Were they popping up around then?

**Simonyi:** Yeah, yeah, that one of course is WordPerfect, Word was kind of coming up but we had more capabilities internally than the peripherals could support at the time. So we were basically ahead of the curve in some ways and misjudging the popularity of the PC interestingly enough in other ways, so we weren't a big factor at the time, but we got tremendous code base, tremendous experience, and then came the Mac. We knew that at one point, you know, GUI will start going. In fact, Bill bought a Star, we bought one of the few Stars that were sold <laughs> and we had a very strange configuration. We had a server, we had a printer and we had one terminal. <laughs> You know, a server was usually to be used for a big installation. Well, we wanted to have one of everything, and basically it certainly wasn't to copy it. My slogan is that I've seen and I made a lot of mistakes in the past. Fortunately, they were proprietary mistakes. That means that under penalty of law you must never do them again. But having that system there was so that people, the new people, could sit down with that system and see what it will be like, you know, what the world will be like, and just get used to it, you know, get the social aspects used to it. You know, you're not going to reverse engineer it obviously, but we had a this funny Star configuration and we knew that some manufacturer will do it. And then so when we heard, you know, Steve was telling us, we had a good relationship with Apple, you know, for example the SoftGuard, the Microsoft SoftGuard, which was a plug-in card into the Apple II, which made it into a Z80 machine that enabled it to run CPM. That was the most popular CPM machine. The CPM market was incredibly fragmented. There were like a dozen different suppliers of CPM machines, and the most popular was the Microsoft SoftGuard, which you plugged into an Apple II.

**Booch:** So Lotus 1-2-3 was a bit of a wake-up call for you guys and with the Mac coming in as well too, is that what then precipitated what became Microsoft Office?

**Simonyi:** Absolutely.

**Booch:** What year would that have been?

**Simonyi:** Well, this was about, we heard about the Mac I think it was '83 that Steve, we went to visit Apple and then Steve came up a couple of times, and Steve is of course very funny, and he has this reality distortion field that is amazing, and I mean I have the greatest regard for him, but I think it helped him too a little bit, because at some point he was going to say, "Now I'm going to show you guys the crown jewels," because we were showing him our thoughts and our vision and we talked about galleries and examples and star sheets and all of that stuff, and he says, "Oh now I've got to show you the crown jewels, this is like top, top, top, top, top, top, secret" and then he shows us or draws us actually, I don't think it was even in the, just draws us a drop-down menu, and you know, we were kind of sitting there and we said, "Oh yeah, Steve, <laughs> that's very nice." Of course, we knew everything about drop-down menus. I mean I've seen drop-down menus for ten years <laughs> by that time. I'm sure it must have been shocking for everybody else, but for us, you know, you kind of forgot that, and he tends to

forget that we all came from the same roots, and some of us were more intimate with the roots than others, but at the same time, you know, he always tells Bill that how Bill stole Mac and then Bill says, "No, no, no, we both stole <laughs> the same, you know, we both stole the Xerox thing." And I mean he's joking when Bill says but I'm not sure if Steve always joked, but it doesn't matter because now they are again, you know, they kissed and made up at the D Conference and they looked into each others eyes and said, "Steve, I love you" and "Bill, I love you too." <laughs>

**Booch:** And on that let's change tapes again. Oh, these are wonderful stories, and we'll continue on with them.

**Booch:** ..Writing some code, but no, not really <laughs>. So we were talking about the incredible growth at Microsoft around that time, you were employee number, in the teens of some sort.

**Simonyi:** Well, I when I was first visited, it was about thirty, thirty-five people, and I was forty-three or so, yeah.

**Booch:** And you grew the team around what became Office, and how large did that grow, how many folks did you come to in all?

**Simonyi:** Well, I probably hired a hundred, hundred and fifty people all over the company, but certainly my team was never bigger than twenty, twenty-five.

**Booch:** So we were last talking about when the Mac came on the scene.

**Simonyi:** Yes.

**Booch:** And the confluence of events of Lotus 1-2-3, the Mac and all of these things.

**Simonyi:** Well, 1-2-3 of course was this big warning, it was a welcome news in the sense that the operating system was doing incredibly well, and--

**Booch:** Because you had this application it was now running on your OS, was that sort of a validation of yours?

**Simonyi:** No, just because the OS was ours <laughs>.

**Booch:** Right, exactly.

**Simonyi:** DOS was ours, so DOS was incredibly popular, on the other hand the applications didn't do well because we were going for the fragmented market. Now, to be sure, when the Mac came we were incredibly well positioned for the Mac for a number of reasons, one of them is that we could immediately port, so that wasn't an issue. The second thing is that the first Mac, and it's not very much appreciated, was a little bit overpowered, so the performance was very high, but the memory was really low, it had 128K memory in the first version, but a lot of that was taken up by the bitmap and by the operating system, so the actual space for the application, was pretty much minimal, and also there were demands on the applications, now we had to have undo, they were quality demands, you couldn't just do a random application, so they were quality demands on the application. And given the extra power we could use our interpreter and cram in the required functionality into the relatively small space.

So just in terms of the technology we were very well positioned, we were also incredibly well positioned in terms of the understanding and the appreciation and just the exploitation of GUI, and which wasn't universal at all at the time, which people were caught up on this was the moment that we were waiting for, for everybody else it was for WordPerfect, it was a surprise or they would have to rethink our stuff. We were doing exactly the opposite, we were ready for this and it's the other stuff that, you know, the simpler environments that we had before, the character enter display, on the IBM PC for example, you could only change the color of a character, you couldn't change the width of the character, and of course that wasn't exactly what wysiwyg was about, you could simulate wysiwyg to some extent. But, so we were really ready for the Mac, or we were ready for GUI, we were ready for-- and the Mac was the first GUI machine. So, we did a bang up job both in Word, and that came together with the first Mac, and--

**Booch:** Was that when it was first named Word?

**Simonyi:** Absolutely, yeah.

**Booch:** And who named it, do you remember?

**Simonyi:** Oh well, I think that marketing named it, I mean we had-- my idea was to call it, you know, Microsoft Multi Tools, and which was used for a while, it was very awkward, and we never thought of Word because you wouldn't protect it, but if you call it Microsoft Word then it's perfectly protectable, there's no question about Microsoft being protected. So everything becomes so much easier to use these generic names like Word and we learned very quickly to defer to-- not quickly enough I have to say, but to defer to marketing in many of these issues. And I joked about it in the past, about for example, I had a big disagreement with Bill about, well, I'll have to reconstruct that, but I wanted to call it Folio when he wanted to call it Pages Number, and then of course the product came out with Folio <laughs>, despite of Bill wanting it to be Page Number. But it should have been Page Number of course.

**Booch:** Bill was right huh?

**Simonyi:** Bill was perfectly right. I was just offering it as a choice I guess to-- I was offering the choice of the audience that one of the people here said it was Folio and one said Page Number, now which one was which, and of course everybody said that Bill, because I was joking about it, so Bill would say, "Folio, no, no, no." I said, "Folio, oh no", I'm sorry, oh now I know what I said was, I said "Oh, we'll tell you three

pieces of data, one of these people wanted it to be Folio, the other one wanted it to be Page Number, and the product came out as Folio, now whose idea was this?" and of course everybody said Bill's idea was Folio because the product came out with Folio, no, no, no <laughs>, my idea was.

**Booch:** Did you find yourself not coming head to head with Bill often?

**Simonyi:** Yeah, well, yes, I think we argued a fair amount I think that he was generally right in retrospect, there was some instance that I'm sorry I wasn't more persuasive about, especially the need of tools that support our development, I think that was the-- Bill wanted to have standard tools very much, and I wanted to have more, if you will, domain specific tools. It's interesting, for example, that just more recently we learned that VisiCalc, for example, was written in a domain specific language, and that it was assembly language but supported with very, very strong macro, and Bob Frankston did a fantastic job in creating that environment, and that explains some of the successes, apart from the fact that it's a brilliant idea, but just the method that it was implemented was quite amazing. So we had a few disagreements in that regard.

**Booch:** Well, I remember one watershed event. I was talking to the current Word team, and they were telling me around the time that Windows 3.1 came out, it was really when it was clear you guys were gaining incredible traction in that market, and Word was an important piece of that.

**Simonyi:** Well, of course, now you're jumping ahead, rightly so, the big deal with-- I mean Mac was the GUI, Graphical User Interface, and how important that is and how amazing it is, and Word and Excel are what quality programs they were. Word, for example, was competing with MacWord, and MacWord was bundled and free, and people, during the trial, I guess people were always asking "How can you compete against a free product?" and funnily enough I don't think that our lawyer ever figured it out, that yeah, there's a very simple way of competing against a free product, you create a better product, and people are perfectly willing to pay for something that they need and that has the stuff that maybe the free product doesn't have. So I think it's perfectly possible to, certainly it's not possible to compete against free with copying the functionality, that's a no-no, but if you offer a better functionality. So GUI being so important it was obvious that what you want to do is get GUI on the most popular platform, which is the Intel platform that Microsoft supplies the operating system for. And so it was obvious for Microsoft to do Windows, and do the applications. I'm not sure if it was 3.1, I think its Windows 95, and that's when it really, really took off, yeah, with the Windows 3 it got very exciting, but it wasn't like exponential, but I think with Windows 95 it got exponential, and by that time the hardware is-- now mind you, the hardware had to get better. In fact a funny line of Steve Jobs, which was true at one point, and it's no longer true <laughs>, and it was also funny at the time, he said that on Windows everything looks worse including a pixel <laughs>, or something to the effect, that even the pixels look worse. And you can say "How can a pixel look good or bad?" and in fact, in the very early machines the pixels on the IBM machine wasn't square, they were rectangular, they had this idiotic, oh sorry, this crazy idea--

**Booch:** Suboptimal <laughs>.

**Simonyi:** Suboptimal idea <laughs> that if you stretched the pixels then certain letters with the same number of bits, you know, a five by six matrix of bits, will be more readable if you stretch them vertically a little bit, the bits. So they went into these pixels that are not square, of course, now to draw a circle

becomes a nightmare, or even draw a straight line is somewhat of a small nightmare, but hell, your letters with the low, super low resolution, which is no longer used <laughs>, is slightly better, okay. And then of course Steve can make his joke how everything looks worse.

**Booch:** Now organizationally this really began the dichotomy in Microsoft where you had the operating system folks and the application folks.

**Simonyi:** Oh that has always been the case, that always been the case, and in fact, if you will, there were also language people, and it was the relationship was somewhat strained. I mean again, when you hear, I'm returning to the trial, when you hear these stories about Chinese wall and stuff like that, not that there should be one, or there is required to be one, but frankly from my point of view there was too much of a wall, I mean I wish things had been as cozy as the other lawyers implied that it was, it certainly wasn't cozy, not because of legal considerations, but-- or maybe it was, and maybe what do I know <laughs>, maybe there was a Chinese wall. But in fact it was difficult to get action either from the operating system folks, or from the language folks, where because we relied on the language people's products to develop our stuff, and gosh, we had special needs. Now, it's funny to say that we had special needs, obviously we weren't special in the-- there were other people in the world that were like our needs, but those people were too few for business to worry about, but if the business has to worry about, well, I mean it would have been perfectly well justified just on selling much better Words, by getting some support.

I'll give you another example, we had a tremendous problem of startup time, and maybe still do, I mean PCs, it's ridiculous, you push a button and then you wait, and every other device you push a button, actually now, TVs are getting worse and worse, which is amazing, you know, I turn on the TV and you wait. But you say Word, or double-click on Word and then it took five or ten seconds to get it going, and for example, when you look at it, much of the time has to do with opening various files. And there's a obvious solution is to have access to the operating system where you can open many files simultaneously, you present the operating system with ten files, <clicks fingers> and then it is done at a single bound, I mean this is Butler Lampson, all of our programs that we ever looked at together with Butler he was always preaching how you have to have these interfaces that can go through the bottlenecks with a lot of bandwidth, you don't go one at a time through a bottleneck, but you package something up, you go through the bottleneck, and then you unpackage it, so the bandwidth is great even though there's a bottleneck, you don't start iterating through a bottleneck. So if the file lookups are bottlenecked, package up like twenty files that you need to open, give it to the operating system and it opens them at once, probably at twice the cost of opening one file, but no more. Now you are ten times faster, ten times faster, okay. And practically, I don't know, it's an afternoon of work, give me a break, all right. No, it's no, because we can't document it, we can't sell it, we can't debug it, we can't do this, we can't do that, come on. And of course the earlier you do it, and I'm talking about now fifteen years ago <laughs>, hello, now it's probably very hard, now I can accept this is the time when the host system was like one-hundredth of the size of what it is today, and I load Windows today and it takes up 20 gigabytes of my machine, now this machine has solid state disk, so I only have 26 gigabytes of disk on this, so I've got 6 gigabytes left <laughs>, just because I put on a operating system, 26 gigabytes, this is all the literature ever written by all the authors, would be a small fraction of, never mind.

**Booch:** With a bit more memory than your first machine, the Ural.

**Simonyi:** Yes, yeah, yeah, that had about 20K bytes of disk, not disk, drum actually.

**Booch:** So you led the applications division of Microsoft.

**Simonyi:** No, development.

**Booch:** Development division.

**Simonyi:** Development, yeah, yeah, yeah.

**Booch:** And before I go on to the time when you moved onto the research side, what surprised you the most, were you surprised by the growth of acceptance of the products you guys produced?

**Simonyi:** Not really, I think that it seemed-- well, you remember this was the early years, and it's true that we were growing exponentially, but we started from small numbers. So, what was really rewarding is when our market valuation exceeded Xerox's market valuation, and I felt <laughs> pretty good about that, that was kind of a nice benchmark, it took about seven years, that basically from call it zero, I mean it's not true, it was an amazing concern, running concern in 1980, it was about ten, twenty million dollars sales a year. But in terms of market value it wasn't that much, and Xerox had a hundred thousand employees <laughs>, and in six, seven years, we were more valuable than Xerox, so that was cool.

**Booch:** I remember the timing you mentioned as well, around this time you became a US citizen too.

**Simonyi:** 1982, that was yeah, '82, and then I started to fly, I decided well, we're going to be successful, I better start learning to fly helicopters <laughs>.

**Booch:** Why helicopters?

**Simonyi:** Oh, well, I thought it would have been a lot of fun, it was available, Seattle was a very friendly town for flying and there were these Vietnam vets that were teaching and this new helicopter, R22, that was the very small and affordable, at least for teaching, I never bought one, but so that was a lot of fun, yeah. We did a lot of informal flying, I mean I sometimes flew to the parking lot of Microsoft and there was a lot of hijinks going on, there were some employees that were setting off homemade explosive devices <laughs>.

**Booch:** Marvelous.

**Simonyi:** Yeah.

**Booch:** So then you moved over to research.

**Simonyi:** Yeah.

**Booch:** And that would've been about what time, what year?

**Simonyi:** I think it would be '92, '93.

**Booch:** Why the move?

**Simonyi:** Well, I think that I noticed the repetitious nature of software development, and I decided that yeah, that was my new calling, that I couldn't contribute that much on participating in the repetition, I wanted to go to one meta level and automate, or at least support with tools the repetition, and that's my obsession with the Intentional encoding, and Intentional Programming started at the time.

**Booch:** And really began around that timeframe?

**Simonyi:** Yeah, that's right, yeah. Yeah, I think, yeah, it's about '92 that happened.

**Booch:** Was that your primary focus in research, or were you leading other research activities as well?

**Simonyi:** No, that was my primary focus, yeah.

**Booch:** So who were some of the team members you assembled around you that were doing work with you, or was it primarily yourself?

**Simonyi:** Yeah, well, oh at--

**Booch:** When you were at Microsoft, research, yeah.

**Simonyi:** At research?

**Booch:** At research, yeah, yeah.

**Simonyi:** I'd like to mention--

**Booch:** Please.



**Simonyi:** Earlier, and Jeff Harbors was a key employee that was responsible for Excel, and he was my first hire at Microsoft, and in fact, he was working for a small outfit, called Apex, at Redmond, and they tried to hire me, and instead I didn't realize that they were trying to hire me, it was a kind of a social meeting that we had and the occasion was kind of a little bit of yachting on Lake Washington, and that they made a pass at me and I defended my honor so fervently that Jeff quit Apex and joined Microsoft as my first hire. And he did a fantastic job and he passed away last year in an airplane accident.

**Booch:** How sad.

**Simonyi:** He was flying a Pilates hot turbo prop, and unfortunately he had an accident with an instructor onboard, and it's an instruction error. But Doc Klunder was another key employee at the early years, and he was and still is a very colorful character, and incredibly talented, Harvard student. At one point we were in a crisis at Excel because he decided to go to California and help the farm workers to pick lettuce, and he just left the job there. And then on the first day all his stuff got ripped off and he got left without any of his money and any of his things, and we kind of had to ship him back <laughs>.

**Booch:** Amazing.

**Simonyi:** And yeah, and Microsoft had nothing to do with <laughs> the assault on Mr. Klunder <laughs>, seriously, but it was, you know, those were the things on which our future were dependent on.

**Booch:** Now you wrote much of the code base of the original Word did you not?

**Simonyi:** Well, oh, Richard Brody, who worked with me at Xerox, for a short time, we hired him, and he wrote much of the code, but in Hungarian he wrote exactly what I would have written, I mean we were almost pair programming it. So it's Richard Brody that wrote the code, but again, once you write down the data structures and everything else the code kind of writes itself, especially if you believe in it, and Richard did, and he's also very, very talented. Actually it's very difficult to find this combination of talent and parallel beliefs, when that happens the power is immense. But usually talent means it's also combined with idiosyncratic beliefs, and to the extent that I have talents I certainly have idiosyncratic beliefs <laughs>, so I'm certainly not an exception to this. But you find multiple people believing in the same thing, and all talented, the results are amazing.

**Booch:** What did you learn most from working with Bill, how did that experience change you?

**Simonyi:** Well, I think that, well I would have to think a little bit, but usually Bill has a number of incredible pithy things that, well at least that affected me, and not necessarily <laughs> others. Other people notice other values in him and he is a very rounded person, he had an incredible upbringing and incredible family, especially his mom Mary, that was just a fabulous person and a very great influence on him. And if you read about his books, he had this epiphany at a young age when he decided that he's not

going to rebel but he's going to study, he's going to express his rebellion through work, I think it was the age of twelve or something, that his parents, he describes this himself, sent him to a psychiatrist, and the psychiatrist convinced him that he is without any challenges because his parents are loving him unconditionally so no matter what he does he cannot change anything about it, so he's basically idling, he better find a more difficult endeavor than to try to tick his parents off <laughs>.

**Booch:** Amazing, marvelous.

**Simonyi:** Well, anyway, that's what he says <laughs>, and you can see that how a kid, and it's completely turned him around, and he decided to find more difficult endeavors. And so, he was a page in the Congress, but he wasn't the page, he was the head page, and he learned the names of all the representatives, of which, I don't know, maybe also the senators, or that goes without saying, I don't know about it, there are 425 representatives.

**Booch:** I think your microphone just fell, if I'm not mistaken, we probably want to click that back up there.

**Simonyi:** Sorry.

**Booch:** No worries.

**Simonyi:** Something was falling, oh yeah, here we go.

**Booch:** Yes. So, returning--

**Simonyi:** By the way, returning to Bill.

**Booch:** Yeah, yeah.

**Simonyi:** He is saying things like one of them is that we are patient people, always is that it is truly by that he means that we are thinking in the long-term, that we worry about day to day but that's not our primary worry, we are worried about the long run. Then he says that profitability is not a natural condition, very little appreciated, and of course when he's, you know, the trial and stuff like that, and nobody understands that profitability is a very, very, very strange concept, and it's a rare <laughs> gift to any organization and any situation that you are actually profitable, the other alternative is much more likely, it means that you really have to, because it's so unnatural, you have to work at, you know, it's not going to stay that way, it's not like a pile of stones that will stay that way, no, this is going to go away unless you work at it and you nurture it, and so on, so forth. So, the profitability is not natural. And then, the one that I quote the most, and probably he denies this, and <laughs> we kind of needle each other about it, is he's sometimes quoted saying "Oh, I could do this in one afternoon in GW BASIC, come on guys." Well, "What's your estimate?" "Oh, it's six months," "Oh give me a break, I could do this in one afternoon in GW BASIC." And of course it's somewhat of a joke, but I always took it seriously, I always said that for every problem there is a means of expression that if you use that means of expression it

would take one afternoon to state that problem, and it's important to say that it's not one second, because I guess, there's a means of expression, the name of the problem, you know, Fermat's last theorem <laughs>, that's one way of expressing it, yeah, but it's not very good. Or Excel for example, you say Excel, no, no, that's not right, why, because it's too short, but if you can say Excel I bet you that I could explain what Excel is in one afternoon.

**Booch:** Indeed, that premise is at the essence of your intentional programming work.

**Simonyi:** Exactly, that's right.

**Booch:** Find the right representation.

**Simonyi:** Yeah, and I always claim that, exactly, enable, first, well, you do it in two steps, one of them is enable, imagine what this representation would be like, and then you enable the existence of that representation, and then of course you have to find the representation, but that's okay, that's a small matter of programming or whatever <laughs>.

**Booch:** So that work fermented from inside research until what time did you decide to leave research, what year was that?

**Simonyi:** Oh, well, I didn't decide to leave research, I decided to leave Microsoft.

**Booch:** Leave Microsoft, yes.

**Simonyi:** Yeah. But it was 2002.

**Booch:** 2002?

**Simonyi:** Yeah, it's almost a decade there, but I did other things too, but the reason it took a while to go out is because of the intellectual property, we really needed to clean up, and when I left we left in perfect terms, you know, with very, very clean, just with a legal framework, and nobody owes, you know, I'm sorry, it's clear who owes who, what, and everybody is very happy.

**Booch:** Yeah.

**Simonyi:** So, and again, it's not an easy thing to do, at the same time it wasn't possible to do what I could call a disruptive technology within Microsoft. I'll give you an example, it would be absolutely impossible to engage a customer from within Microsoft with a disruptive technology, it's not good for a customer, it's not good for Microsoft, it's not good for the organization that is doing the engaging, you

wouldn't learn anything, all the hell would break loose, everybody would get confused, it cannot be done, it shouldn't be done.

**Booch:** Unless it made perfect sense to move on.

**Simonyi:** And at the same time you cannot do a good job without engaging some customer, so this created this-- and that was pretty clear to Bill, it was pretty clear to me too, so it's not that there's anything wrong with Microsoft, it's just not the structure where to do something like that.

**Booch:** Right, and thus was born Intentional.

**Simonyi:** That's right. And okay <laughs>.

**Booch:** So tell me a little bit about where you're trying to head with Intentional, what would you give me as an elevator pitch to Intentional's intent?

**Simonyi:** Right, well, I think that what we are trying to do is bring the subject matter experts directly into the production process. So, we have different pitches for the management, for the software engineer, for the management we say "Look, the most value in today's software comes from the subject matter, it's no longer a programming issue, and if you want to automate your hospital, it's not a matter of software, it's a matter of expertise, of what's going on in a hospital." The machines are perfectly capable of handling two thousand patients or ten thousand patients, a hundred thousand patients is no longer a big deal. The issue is what are the situations, and who has that knowledge, the subject matter experts have that knowledge, certainly the programmers don't have that knowledge. And so to capture the subject matter experts' contributions, and make them a direct contributor to the software, that's what we are promoting, and we are promoting it with a specific tool, called the Domain Workbench.

**Booch:** Yes.

**Simonyi:** So it's the Domain Workbench to me is the holy grail of domain oriented development. There is a lot of domain orientation now is a very normal thing, people talk about DSL's, people talk about MDA, MDD, Model Directed Development, of course UML, and--

**Booch:** I've heard of that <laughs>.

**Simonyi:** Yes, it's all domain orientation is key.

**Booch:** Yes.

**Simonyi:** So, we are supporting it with a Domain Workbench, that is a universal tool that let's domain experts and others express things in arbitrary notations and with a legacy that is protected in a changing environment, and which then can be easily transformed into running software by the programmers.

**Booch:** And it's interesting to see the threads of this through your life, I mean the meta programming work you've done, the very domain specific things you observed, even the earliest days you were programming.

**Simonyi:** Over the earliest days I always thought when I looked at those equations in the process control system, and that was clear that look, my problem was to encode the equations, the equations were created by the process engineers. And so, even working with the wireless, and encoding the hardware engineers' contributions in a form that is familiar to them. And if you look at [Jerry] Weinberg, for example, the book on "The Psychology of Computer Programming," there are examples in there, and we are talking about 1974--

**Booch:** Right, he was so ahead of his time.

**Simonyi:** Yeah, there was examples in there where, for example, assembly parts, parts of lists for a car, were created in a domain specific manner.

**Booch:** Marvelous. We need to change tapes and we'll finish on the last one, and we'll hear about the space.

**Simonyi:** Great, I'm looking forward.

**Booch:** You are just such a fascinating individual, Charles.

**Simonyi:** Blush, blush, blush.

**Booch:** So I want to hear a little bit about the non-software piece of your life. You've had your love for flying. You got your helicopter license, and then you got your pilot's license for multiengine aircraft.

**Simonyi:** Multiengine and, of course, IFR. Very important. Now, I'm type rated for the 7X.

**Booch:** But then, of course, you went to space. Tell me about your dream to do that.

**Simonyi:** Well, the interesting thing is that there was no dream. It just wasn't in the cards, was it? I mean, this whole idea of space tourism is an amazing development. And probably I think that suborbital tourism is very realistic, but let's just focus on orbital tourism. I don't think it will-- I think it will disappear for a very long time. I don't think there'll be tourists for the next decade. There will be maybe two more.

Richard Garriott is the next one. And, you know, he's a computer person. He's a game designer. His dad is an astronaut.

**Booch:** He'll be the third, because Mark Shuttleworth...

**Simonyi:** No.

**Booch:** ...Shuttleworth was the first programming type.

**Simonyi:** Programmer type. That's right. Mark Shuttleworth was-- he was the second-- the first programmer and the second space tourist. So Eric Anderson is the head of Space Adventures. And it was his idea to go to the Russians, who were very cash strapped, we are talking about 2000 and so on, and see if they would do-- he thought that he would be able to sell the seats that are available. And I think that he optioned out seats well into the future. And then he was very, very successful in marketing them. And, of course, Dennis Tito was the first who flew under very difficult circumstances, very much opposed by NASA. There was a famous confrontation in the gates of Houston Space Center, where the Russian crew showed up, you know, under their contract and all that and-- including Dennis Tito. And NASA said, "The Russians can come, but Dennis cannot." And then the Russians refused, they say that, "If he doesn't come, then, you know, we're a team. We have to go together." Of course, they had their incentive because they would be getting the money, the organization, of course, not the individuals. And then NASA relented after a the standoff for a day or something, where these guys were cooling their heels at the security office, which completely changed, by the way, the situation.

I felt very welcome by NASA. And NASA has become accustomed to the idea and appreciated the importance. It's kind of complicated going into it. But there are all kinds of political issues -- it's actually quite difficult for the U.S. Government to pay the Russians for services that the Russians provided. There is this Iran Nonproliferation Act that caused a lot of problems. So being able to funnel funds to the Russians and keep the program alive, which is critical to the success of the international space station, you know, when, for example, our shuttle wasn't flying, the Soyuz was the only way to support it. And, of course, the Soyuz, name of the Russian spacecraft, is the lifeboat on the ISS, International Space Station. So every American or international astronaut who stays on ISS has to go through training so that they can utilize the lifeboat if that became necessary, which hasn't happened. But, you know, there's a real danger. So there's this extra seat. When you change the spacecraft every six months-- you have to rotate the spacecraft. It has a finite lifetime. And you have to rotate the crew, because six months is quite a bit of time in space. It's not a benign environment as people might have hoped that it would be. It's a difficult environment. And during that rotation, there's an extra seat. It's called a taxi seat. There is this opportunity. And I actually went and visited Baikonur as a tourist to witness a launch, and Space Adventures organizes these tours. They are, again, kind of difficult, because access to Baikonur is difficult politically. It's a Russian base on Kazakh soil. It's accessible only by, you know, it's under all kinds of treaties. It's not like the Russians don't want to be open. Once you get to Baikonur, the openness is unbelievable. In the first time I was there, we were partying, two hours before launch, right next to the vehicle. And fully fueled. I mean, you can see the snow on the skin. I tell you the skin is not farther from me than the camera is right now. It's about ten feet away, and it's all kind of smoldering there. And you see the astronauts coming and going up the stairs. And you see the launch. The second

time I saw it, from very, very close up, in a completely illegal position. It's amazing. And Eric kind of raised the possibility that I might want to do that.

**Booch:** What year would this have been that he was proposing this?

**Simonyi:** Maybe 2004 he raised it first. I didn't want to believe that I would be capable. Just, you know, I am in good health, or I thought I was in good health. And I exercise regularly. But, you know, it seems pretty extreme. And I wasn't exactly sure of what is necessary. And frankly, when I saw it, it frightened me, I mean, just the act of launch. And, you know, I got kind of physically excited. And, you know, not necessarily fearful but just tense and, well, okay. So maybe I was a little bit fearing for the other people. And I couldn't quite see myself being in a position as participant being more difficult than observer. Well, turns out that's not quite the case, but I didn't know it at the time. But I was encouraged. And I decided that, well, so is hard to make a commitment like this. At least for me, it was to do it slowly. And I made a small series of commitments. For example, I asked if it were possible to train with the trainers, for about a couple of weeks, at a neutral place. I never seen a Russian trainer. I didn't know what materials do they use or what do they show you, what are they asking you to do and so on. So in Palma Majorca, we stayed two weeks with four trainers, including a Russian teacher, who also worked as a translator and went through some of the training materials, which was amazing. And one of the cosmonauts, Yuri Usachev, came and showed slides. And we tasted the food and learned about 600 technical words in Russian just doing that process. And I was very impressed by their training process, but looking at Yuri's slides of the space station and the spaceflight, that was still way out. So the next, let's do the medical stuff. I was worried a little bit about accelerations and maybe weightlessness. You know, do any of these things disagree with me? I had a fair amount of-- I have about 300 hours in high-performance jets. So I know what gs are like, but I was always wearing g-pants. And I knew that without g-pants I end up-- in space, you don't use g-pants. So I wondered what that is like. I didn't want to believe that it's not a problem. In fact, it isn't a problem. The acceleration is called plus Z, instead of plus Y. So the blood goes transversely your body, which there's not many places to go, instead of going from your head to the toe, which then causes you to have tunnel vision and then lose consciousness. So we went on a centrifuge.

**Booch:** Tried the Vomit Comet?

**Simonyi:** Yes. I didn't try Vomit Comet, but I did ride a commercial version of that from ZERO-G Corporation. We rented the whole plane, and I took 20 friends of mine on it.

**Booch:** Marvelous.

**Simonyi:** Yeah, two of them got sick, but the others just enjoyed themselves tremendously. I mean, Zero-G flight-- we had about ten parabolas. And they actually do a good job. You don't go into Zero-G right away, but they go into lunar and Martian gravity. I flew Martian gravity and then lunar gravity and then Zero-G. And lunar gravity is fantastic because there you still know up and down. And it's familiar, except that you are incredibly strong. So for example, you can do-- you can tumble very easily. You can jump, for example, jump about 10, 15 feet into distance and land on your hands and then roll, make a roll. Those are like Superman gestures. And they're very natural because of gravity being so low. Now, Zero-G is something different. Because there the-- you have to get used to the fact that you cannot, well, you

have to hang on to something. Otherwise, you are not-- it's easy to be ineffective. They tell you, for example, not to try to swim. And, of course, everybody's swimming. It's completely useless. It's not like it hurts you to swim. But it's completely useless, and you're wasting the time. You know, you could do other stuff. So we did that. Then I did all the doctors again in the U.S. Go through all the tests. I was over 50, and I was supposed to do all these things, checkups that I didn't do. And this seemed like a great excuse to get going. And I saw a lot of-- I went to Galveston, to University of Texas Medical Branch, and then Houston of course. And I just saw a zillion doctors. It wasn't all-- turned out that I was very healthy. But I did fail a couple. Of course, I have a tremor in my hand that's an essential tremor. No Parkinson's. I know because that's the first thing that they suspect and they want to make sure that you don't have it if, I mean, if your hands shake. And I, also, have a heart-- just a rhythm that is...

**Booch:** Arrhythmia.

**Simonyi:** It's kind of arrhythmia. It's PVC, Pre-ventricular whatever it is F, no, C.

**Booch:** Fibrillation? Oh, C.

**Simonyi:** No, C, PVC. Maybe that's polyvinyl chloride. Contraction! Pre-ventricular contraction.

**Booch:** But amazing that your whole body was checked from top to bottom.

**Simonyi:** It was perfectly OK. So for example, the-- on an American centrifuge, they kicked me off right away. They said, "Oh, this guy has PVCs. No centrifuge." But we went through the British centrifuge.

**Booch:** They have different rules.

**Simonyi:** No. They didn't worry about this-- they weren't worried that, suppose I died, you know, I'm going to sue them. Everybody worries that I-- gosh, you know, I do have-- insurance from the American government wanted me to insure the government, or, NASA wanted me to insure the government so that my estate is not going to sue the government for damages, okay, in case I would die or get injured in the American segment of the space station. Now, it sounds really funny. But it costs real money to get such insurance. And, again, something that I said, "Can't I just sign a waiver?" No. That would be too easy or too cheap or something. The funny thing is that it's not the government that gets the money, it's the insurance company, for this completely ridiculous event, okay?

**Booch:** Right.

**Simonyi:** So if it went to the government, I would have said, "Great. I'm happy to contribute to NASA and the government, to their good works." But, no, have to contribute to Lloyd's of London because of this idiotic nonsense. And, you know, I wonder if, well, I make stupid arguments like, "How come you don't accept my signature on my waiver but you accept the insurance company's signature on their promise? Why can't they just say, 'Oh, we didn't mean it'? If I can say, 'Oh, I didn't mean,' to my waiver,



why can't the insurance company say?" And not like, I mean, the insurance company's not that much bigger. Well, I guess they are. But the names there that are supporting it are not, I mean, it's not like I don't have the funds to make good my promise. But I think that the difference is that they are alive and I am dead.

**Booch:** So after the medical test, what were the next steps to creep you closer?

**Simonyi:** So anyway, the medical tests weren't, like, completely okay. But at every point, every time we failed something, which was very seldom, but we did fail something, the Space Adventure said, "Oh, that's not a problem. You know, the other guys failed too, and they got fixed." And it's amazing what the other guys did. Richard, who hasn't flown yet-- but he showed me his scar. He had some minor condition that-- and I'm not going to go into it but-- which had to be fixed or should have been fixed and-- which he was aware but was never fixed or whatever. But he got a scar, you know, across his belly, like, this big. I said, "What? They don't do laparoscopy?" "No, not for this." So, yeah, and certainly, yeah, some of the other-- so there were quite a few medical procedures done to the others too. I didn't have that. But certainly...

**Booch:** Little fine tuning.

**Simonyi:** ...little fine tuning, yeah. So then I had to do the medical in Russia, which was getting-- then we were getting-- actually, I think, by this time, I-- that's right. Then I signed up. You have to sign up, because otherwise the seats would be gone. So then I went to Russia. Also, the Russians wouldn't give you the medical or whatever if we hadn't signed up. So I'm signed up. I'm committed to go to Russia, get the medical there, do well, at least as far as they're concerned. I felt, you know, after some of the tests, I was you know, there's this thing called a vestibular chair, where they are rotating you. You move your head back and forth, and it creates a Coriolis acceleration in your inner ear. And it confuses you completely. And it helps you tolerate weightlessness much better, and I believe that it does. And toward the end of my training, I got really used to it. But the first time I tried it I wasn't a happy camper. And then even when I went to my hotel room and we went outside, on a hot day, you know, I got a little bit sick just-- not literally but just this is not easy. And I was wondering should I do this. And then I got accepted by the medical commission. And when I got accepted, the-- I could see the other crews. There were some Russian crews. There were Malaysians there. Governments are getting into the action, because if individuals can buy the seats governments can buy the seats. So Malaysia bought the seat. Korea bought a seat. And then they decide within the country of whom to give it to. And they run a contest or whatever. So the Malaysian guy just flew last September, and the Korean guy's going to fly this April. But, you know, I could see, wow, these seats are filling up. And then Eric says, "Charles," we were talking about, you know, flying in '08, "unh-unh, you have to fly in '07, you know. Because you are now-- you are the only one who's qualified to go for the next one." You know, because I got some training. And we had been training with everybody else. And I got the medical. And we got to grab the seat. And so I was thinking about it and looking at the other people and said, "Let's do it. Let's do it. Let's interrupt -- everything I do and basically move into a monastery in Star City and take eight months off." People are remarkably understanding. You don't basically break all your promises, all your appointments.

**Booch:** You drop out of life.

**Simonyi:** But say, "Hey, I'm going to space," and people say, "Oh, in that case..." You know, either I have cancer, you know, or I go to space.

**Booch:** It's a reasonable excuse.

**Simonyi:** It's a reasonable explanation, reasonable excuse. And my coworkers, you know, at Intentional were very nice about it. And so that's how-- in September, I moved into Star City and started training.

**Booch:** And the launch was when?

**Simonyi:** April 7 of 2007.

**Booch:** Tell me how you felt that day as you were walking to the spacecraft.

**Simonyi:** How do you feel? I couldn't imagine that I would ever feel that feeling. And it turns out that it's done very well. It's so well choreographed. It's so natural, and here's the difference between the spectator and the participant. As a spectator, I was nervous. I was maybe even fearful a little bit, you know, get clammy. As a participant, it was a wonderful experience. And the reason is that we know so much more about what's going on. Basically, all we did was practice it, practice every detail, not necessarily at the same time but so much so. I mean, give you an example. Two days before the launch, we did go through the just dressing up, giving a press conference, getting a pressure check, and then we just walked to the spacecraft, that was still in the building, and going into the spacecraft and just sitting in it and coming out. And then we came out, which was not standard, exactly in the same area, the same building. And I remember, for example, I had to put on my underwear. And I noticed that I was given two sets of socks, you know, long underwear, and then there are these black socks. And so I thought it was a mistake. So I put on one-- do one sock and put on the long underwear. Then I go to the next station, where we're going to put on the spacesuit. Because I know exactly how to put on a spacesuit. And the crew chief, you know, looks at me and said, "Where are your other socks?" And I say, "What other sock?" You're supposed to wear two socks so that we can take off one of the socks and the other sock underneath is clean so that when it goes into the spacesuit it's a clean sock, okay? I didn't know that. And they, "Oh," you know, <makes mumbling sound>. But nobody told me. But it was during the rehearsal, okay? So, yeah, yeah, we kind of brushed my sock, you know, to an extent that we could. I was just walking, you know, 100 feet in a clean area. Never given it any thought. But the procedure is that you wear two socks. And then they take one of the sock off as you go into the spacesuit. So by the time I was at the last day, everything was so calm. You know, the previous day we watched a movie. That is a tradition. It's always the same movie, The White Sun of the Desert, which is called an eastern, you know, as opposed to a western. But it's exactly the same genre, you know. It's a lot of fun. And the friends come and visit you. Like in a prison, you are behind glass. And then you have about...

**Booch:** Quarantined.

**Simonyi:** You are quarantined. You have about 30 minutes with them. And then, on the day of launch, we launched at 9:00 p.m. but this was about maybe 5:00 or 6:00 p.m., a DVD-- they got a DVD going and played greetings unbeknownst to us from the family.

**Booch:** How sweet.

**Simonyi:** Yeah, it's very well produced DVD of, you know, my mom and even my-- the taxi driver that always drives me, in Budapest, he was sending me a message. It's unfortunately not from America, that would be too far, but from Hungary. And then the-- all the all other traditions, you know, the urinating on the tire of the bus, which is-- actually, it's a quite nice tradition.

**Booch:** It's practical.

**Simonyi:** It's practical. I didn't know if I would participate in it. But then I realized that the same crew chief that just sealed my suit was encouraging me to do it. So he's going to reseal the suit just the same way as in the building. So, yeah, let's do it. And then the other reason is that all the support group can now get out of the bus also. They're just standing around having a smoke. Because this is the, you know, they don't have a chance of getting a smoke. So in a sense, it's also a smoke-- while we are urinating, they're-- they are having a nice smoke. But also means, you know, it means, look, I mean, I'm not in favor of smoking mind you. But I think there's a mental message there that, look, everything is fine. There's no hurry. It's a normal thing. We are just on our way to the pad. And we just pee on the tire. They're having a smoke. And then, you know, oh, and the other thing is that, the commission, that we presented ourselves at the building, the original building, they were also driving to the pad in a different route, on a less direct route. So by the time we arrived to the pad, the commission was there waiting for us again, which was also very nice. So this was also way to delay us just so that, the other people, they can get there. It's all very practical. And everything goes-- the Russians have this wonderful saying. It says, "<speaks Russian>," means war is war, but dinner is always on schedule, or, by the schedule. It's by the schedule. So no matter what happens, you know, there's a schedule that you follow. And I don't know. It's really true. They don't know what a hold is. There's no such thing. There's a reserve day in their schedule, which they practically never use. But they don't have a hold. It's kind of this inexorable movement. And in all the three cases when I was there, there was never a second of delay. Now, of course, everything went very slowly in some ways. You might think that it's going very fast, because, again, two days before launch, the vehicle wasn't together yet. You know, it's put together, like, the day before. And then the train-- it's put together right on the top of a train. And then the train goes out to the pad. And then they straighten it up, and then the people go plug in the plugs. And now it's ready to go. It's a simple machine that has been working for the last forty years.

**Booch:** It just works?

**Simonyi:** It just works. It's not stressed. So machine is not stressed. The people are not stressed. Was really the calmest thing. I can't tell you how calming it was. We were in-- already in a-- they were playing ABBA in, you know, for five or ten minutes. They said, "You guys want to listen to some music?" And I said, "Sure. I'm not against, you know, whatever commander says." And then they were playing ABBA.

**Booch:** So when you were in space looking down at the earth, how did that feel?

**Simonyi:** The fact that you are there and it's-- it is amazing. And you are with five other people and you are only six people in space at the time. And the size of these six people that are in space. It's a privilege, you know. You are grateful for your good luck that you have done it. And the sensations are amazing, the-- just the view. This is Low Earth Orbit. So you're not-- people always talk about the fragility and the smallness of the earth. That's probably true the farther you get. And probably from lunar distances, it's-- it must be something quite extraordinary. From Low Earth Orbit, it's enormous. It's enormous. It looks like a giant sky. That's how I can describe it. It's blue. It's blue as the sky. And it goes from, you know, the right time of the day or your orbit, it goes from left to right everywhere. And when you look at it close, it's not that different than looking from a high-flying airplane, except I think that the earth is like a fractal. Nature has this fractalness to it. Maybe it's the-- it depends on the scale. It looks the same. And I sometimes have thought I was looking at a lake that has waves on it. And then I realize that it's not a lake. It's, you know, it's the Mediterranean. And those are not waves. They are tides or, you know, patterns that repeat on the order of tens of miles rather than, you know, hundreds of feet. But otherwise it kind of looks the same. Much of the earth is uninhabited, and you don't recognize where it is. If you just happen to look out at some random time and you see what's below, you wouldn't know what it is.

**Booch:** But at night, you see the lights of the cities too, don't you?

**Simonyi:** In nights, you see the lights of the cities, weather permitting. You see lightening. And sometimes, you know, for example, the U.S. looks to me like a brain, like the synapses. The cities are the synapses. And there are these tendrils that go...

**Brooch:** The roads.

**Simonyi:** ...the routes between the cities and the strip malls on the routes or whatever lighting them up. They look like these thin, thin connections between the synapses. So I made a note about that, just that metaphor, up there because it was so striking. Again, it's the-- it is the fractal nature. And there is sense to it, of course, why the pattern is-- that pattern is a good pattern. You know, it's a good design pattern. It should be used. Yeah, I mean, you, too, have these concentrations, and then you have the communication between the concentration. And then, by a given stain, if you will, if you stain it in a correct way, then you can make it visible. In this case, the stain, in the positive sense, don't misquote me, is the lighting visible at night.

**Booch:** So from postwar Budapest to space. What an amazing life you've had during that time frame. Project out for me. The next generation, what are they going to see? What do you think the world's going to be like for them?

**Simonyi:** I think it will be somewhere in-between the-- kind of the vision where everybody is immortal and the total disaster. I think it will be-- in many ways, it will be just like-- in many ways, the world is not going to change. It's amazing. I love to look at old magazines, you know, Time magazines from the '50s. Very few products and services are recognizable from that time. But for example, the Cessna 150 is

perfectly recognizable. It's not called the Cessna 150 anymore, but it's called 175. But I sometimes hear on the radio 150s flying too. But it's essentially the same gizmo. So some things will be exactly the same, and we'll probably worry about the same stuff. Other things, there will be this incredible jumps. And that was, for example, in computers. That was truly unexpected, the extent of it. I mean, even if you hear about Moore's Law, that's great. But for example, that disks would go according to Moore's Law, that was completely unexpected. I understand Moore's Law. I understand why smaller is less power, smaller feature size. But wait a minute. That works for disks too? And, of course, bandwidth with the optical fiber having that richness in bandwidth. And, of course, now with the lithium batteries, we get even that a little bit more under control. I hear, at Stanford, that they are looking at the factor of ten more in battery technology.

**Booch:** Digital paper.

**Simonyi:** Oh, well, yeah, digital paper, the display, would change the display technology. You know, my dream is, of course, that software would cease to be the bottleneck in this-- the digital horn of plenty, which it is today. And I think we'll be able to do that. To me, that's the biggest open opportunity, the obvious opportunity.

**Booch:** That's where you're focusing at this point in your career.

**Simonyi:** That's why I absolutely focusing on it, and that's why I continue to focus on it.

**Booch:** What a great phrase, the digital horn of plenty. I like that.

**Simonyi:** And there's a bottleneck on it.

**Booch:** That's right. There's a bottleneck on it.

**Simonyi:** That's right. But it is a horn of plenty. There's no question.

**Booch:** Charles, this is wonderful. Is there any bit of your life that I've missed here that we ought to talk about?

**Simonyi:** No, not that I want to speak about.

**Booch:** We covered all the right things.

**Simonyi:** Yes.

END OF INTERVIEW