

von Ronald Bieber

Streben nach Wissen: Pascal, Modula und Oberon

OCG Generalsekretär Ronald Bieber hatte im September die Gelegenheit, ein ausführliches Gespräch mit dem Informatik-Pionier Niklaus Wirth zur führen. Begleitet wurde er dabei von Jürg Gutknecht. Input zu den Themengebieten kamen von Wolfgang Pree (Universität Salzburg), Hanspeter Mössenböck und Gusatv Pomberger (beide JKU Linz). Lesen Sie auf den folgenden Seiten das Gespräch, das am 16. September 2019 in Zürich stattfand.

Herr Professor Wirth, vielen Dank für Ihre Beiteitschaft zu diesem Gespräch. Sie haben nach Ihrem Studium auf der ETH Zürich den Master schon im Ausland, in Kanada, gemacht und haben dann Ihre Promotion als Fulbright Student in Berkeley absolviert. Welchen Einfluss hatten diese Auslandsaufenthalte jetzt im Nachhinein für Ihren Werdegang?

Wirth: Das kann man nicht unterschätzen, das war prägend. Ich bin gleich nach dem Studium ausgewandert, und zwar mit einem regulären Auswanderungsvisum, oder Einwanderungsvisum muss man ja sagen. Weil da der Vietnamkrieg tobte, bin ich zuerst nach Kanada gegangen, damit ich sicher nicht eingezogen werde und ich habe mich da schon für Computer interessiert. Ich habe einen Kurs in numerischer Mathematik besucht, aber jedes Mal, wenn eine Übungsaufgabe gelöst war und man zu den Computern gehen sollte, waren die Techniker dort am Flickern. Wir haben das ganze Semester keine einzige Aufgabe am Computer gelöst. Das waren eben noch die Zeiten mit den Elektronenröhrenrechnern und die waren sehr unzuverlässig.

Ich wollte eigentlich in Amerika arbeiten, als Ingenieur, aber man hat mich dann ermuntert nach dem Master Degree weiter zu studieren und so habe ich die Gelegenheit benutzt, um in Kalifornien - das ich ohnehin einmal besuchen wollte - weiter zu studieren. So bin ich nach Berkeley gekommen. Dort habe ich auch ein Teaching Assistantship gehabt und nach zwei Jahren ein Stipendium.

Gibt es da einen bestimmten Namen, weil Sie gesagt haben „Ich wurde ermuntert weiterzumachen und nicht arbeiten zu gehen als Ingenieur“, dem man jetzt ein herzliches Dankeschön sagen kann?

Wirth: Nein, eigentlich nicht. Es war ein Department, das gerade an der Universität gegründet wurde und wir waren etwa fünf Graduate Students. Ein Brasilianischer, ein Franzose, eine Chinesin und noch jemand und ich. Es war eine internationale Mini-Gesellschaft. Dort habe ich als Masterarbeit einen Analogcomputer gebaut. In das Computing-Gebiet bin ich erst in Berkeley wirklich gerutscht.

Und diese vier, fünf Studenten: Haben die Sie in Ihrem späteren Werdegang begleitet?

Wirth: Nein, nein. Wir waren einfach während des Jahres Kollegen.

Sie haben in Ihrer Karriere die Entwicklung der ganzen Computertechnik und auch der Informatik schon sehr früh begleitet und geprägt. Können Sie vielleicht drei oder einige Punkte herausgreifen, die Sie besonders beeinflusst haben oder die für Sie ganz wesentlich und wichtig waren?

Wirth: Im zweiten Jahr in Berkeley kam der holländische Professor, [Adriaan] van Wijngaarden auf Besuch und der hat mir einige Ideen geliefert, was man da unternehmen könnte. Er hat mich sehr beeinflusst. Durch ihn bin ich in die IFIP Working Group 2.1 gekommen. Davon vielleicht später noch etwas mehr. Der Professor, bei dem ich offiziell war, war ein Pionier im Hardwarebau und wir haben uns nicht sehr oft getroffen.

Ich habe dann durch Zufall eine Gruppe von auch Graduate Students getroffen, die haben in einem Keller in einem Nebengebäude gehaust und etwas gemacht. Das war damals sehr neu: Die haben einen Compiler gebaut und korrigiert. Das war damals etwas ganz Neuartiges, für einen Dialekt von Algol 58. Das hat mich fasziniert, und zwar vor allem, weil dieses Compilerprogramm in der Sprache selbst geschrieben war, das es übersetzt. Man hat da intensiv dieses Bootstrapping-Prinzip anwenden können. In der ganzen Gruppe war nur eine Frau wirklich kompetent und verstand dieses furchtbare Programm. Das war wirklich sehr, sehr schwierig zu verstehen und zu durchwühlen. Das Übliche, kann man vielleicht sagen, Software-Chaos. Auch heutige Software ist oft noch sehr undurchsichtig. Das war 1961.

Also während Ihrer Promotionszeit?

Wirth: Ja, genau. Ich habe nebenbei in diesem Gebiet promoviert.

Sie waren in die Algol-Entwicklung involviert.

Wirth: Das war die Zeit, wo der Algol-Report erschienen ist und man ihn studiert hat, auch in dieser Gruppe und so bin



Niklaus Wirth im Gespräch, alle Fotos: Ronald Bieber/OCC

ich auf Algol aufmerksam geworden. Ich habe gefunden, dass dieses Gebiet, das sich in solch einem Chaos befindet, das wäre doch ein lohnendes Thema für eine Dissertation, da etwas mathematische Ordnung hineinzubringen! Das hat dazu geführt, dass ich dann dieses Gebiet beachtete habe.

Durch Ihre Mitarbeit an Algol haben Sie dann die Idee zu Pascal gehabt?

Wirth: Ja, das kam aber später. Wenn Sie wollen, kann ich Ihnen schon etwas über diese Geschichte erzählen. Ich bin durch den holländischen Professor [Adriaan van Wijngaarden] in diese Working Group hineingekommen, mit dem Ziel einen Nachfolger von Algol 60 zu definieren. Das hat man damals noch als erstrebenswert und relativ eilig empfunden. So bin ich einige Male nach Europa geflogen zu solchen Meetings. Da hat sich dann relativ bald eine Spaltung dieser Working Group ergeben. Die einen wollten noch einmal ein Monument wie Algol 60 kreieren, also etwas, das wirklich innovativ und neuartig war und in das man neue Ideen einbrachte. Wijngaarden hat ein Projekt mit dieser Zielrichtung vorgelegt. Ich selbst war etwas bescheidener. Ich habe gefunden, dass es relativ rasch etwas Praktisches braucht, vor allem in Konkurrenz mit PL/I. Daraus ist dann eine Sprache entstanden, die wir sofort in Stanford, wo ich dann hingekommen bin, implementiert haben. Das wurde dann

als Algol W bekannt und vor allem auf IBM Computern benutzt und verbreitet.

Aber in dieser Working Group ist man dann irgendwie zu keinem Schluss gekommen und hat sich sogar aufgespalten. Ich habe diese Working Group dann auch verlassen und meinen Algol W Vorschlag weiterverarbeitet und daraus wurde Pascal. Das wurde 1970 publiziert.

Wofür steht Pascal?

Wirth: Das habe ich zu Ehren von Blaise Pascal

gewählt. Erstes war er der erste, der solch ein Gerät gebastelt hat, das man heute Digitalrechner nennen kann. Es konnte addieren und hatte Komponenten, um dem Vater bei der Steuereintreibung zu helfen. Er war damals 16 Jahre alt. Zweitens brauchte ich einen Namen, der weniger als sieben Characters lang war, um auf der CDC-Maschine damit arbeiten zu können, aber das ist Nebensache.

Es folgte dann Modula, Modula 2 bis zu Oberon.

Wirth: Pascal war ja eine Sprache ganz im Sinne von Algol, also ziemlich kurzgefasst und dicht. Algol war ja vor allem numerisch orientiert und man wollte den Anwendungsbereich erweitern, aber es blieb eine Sprache für monolithische Programme, also in einem Stück. Das Computergebiet hat ja rasante Fortschritte gemacht, hin zu Großcomputern, die große Programmsysteme verarbeiten konnten und da war es wichtig, Teams zu haben, die man in einzelnen Stücken entwickeln konnte, die man nachher zusammenfügte, also das Konzept des Moduls. Modula ist eigentlich Pascal mit dem Modulkonzept und noch einigen anderen Dingen. Um Pascal teamfähig zu machen, ist Modula entstanden.

Oberon ist im Vergleich etwas objektorientierter?

Wirth: Das ist dann wieder später. Pascal 1970, Modula 1979 und Oberon 1988. Ich

habe mit der Zeit schon eingesehen, dass Modula zu kompliziert war. Es ist den Weg aller Computersprachen gegangen. Immer komplizierter. Es ist einfach viel zu leicht, immer etwas anzufügen. Das wollte ich korrigieren und so habe ich dann zusammen mit Jürg Gutknecht Oberon entwickelt, das viel dichter und viel kleiner ist als Modula. Aber ich glaube man kann sagen, dass es nicht weniger mächtig ist.

Ich glaube, das war eines Ihrer Mottos: „Make it as simple as possible, but not simpler.“

Wirth: Das „make it simple“ muss man bei Oberon noch ein wenig erklären. Aber ich weiß nicht, wie ich das erklären sollte. Der Spruch stammt ja von Einstein und die Meinung ist natürlich schon, dass man da nicht zu weit gehen sollte, dass man mit der Sprache gar nichts mehr Seriöses anfangen kann.

Gutknecht: Es war dann schon noch die Type-Extension, die halt doch ins Laufzeitsystem übergriff. Sie haben gesagt: Objektorientierung und das haben wir eben reduziert auf die beiden Basiskomponenten, die nötig sind, um Objektorientierung einzuführen. Das ist einmal die Prozedurvariable, also die Variable mit Prozedurtyp und das zweite war die Type-Extension, Subclassing nennt man das bei der Objektorientierung. Das war eine große Welle, dieses Objektorientieren. Oberon ist nicht nur kleiner geworden, sondern eigentlich mächtiger. Das ist, was wir besonders hervorheben.

Sie waren in dieser Zeit zwei Mal im Ausland, bei Xerox in den USA. Hat diese Entwicklung Ihre Arbeit dort auch beeinflusst?

Wirth: Ja, absolut. Ich glaube, das war sogar noch wichtiger als dieser holländische Professor. Das hat bei mir wirklich tiefgreifende Einsichten gebracht und hat mich auch wieder dazu gebracht Hardware zu studieren und zu bauen. Ich habe Elektroingenieur studiert, aber in diesen vergangenen 15 Jahren wurde da alles anders. Gut, das Ohm'sche Gesetz, das galt immer noch, aber anstatt Röhren brauchte man jetzt Transistoren und integrierte Schaltungen, Chips. Und da habe ich mich dann während dieses Jahres

und vor allem danach eingearbeitet. Das war das Jahr 1976/77 bei Xerox und dann noch einmal 1984/85. Die Sprache Oberon ist eindeutig aus jener Zeit hervorgegangen. Zum Teil unter dem Einfluss der Sprache Mesa und Cedar, die dort intern entwickelt wurden.

Sie haben das Ohm'sche Gesetz angesprochen, das beständig bleibt. Bleibt auch ein Wirth'sches Gesetz beständig? Die Software verlangsamt sich schneller als sich die Hardware beschleunigt. Können Sie da noch etwas dazu sagen?

Wirth: Da muss ich auch ehrlicherweise beifügen: Das habe ich das erste Mal von Martin Reiser gehört, diese Aussage, aber es stimmt durchaus.

Stichwort Programmiersprachen: Würden Sie sehen, dass heute konzeptionell schon das Ende der Fahnenstange erreicht ist?

Wirth: Die Entwicklung geht weiter, aber es ist vielleicht das Ende der Stange, so wie wir sie wahrgenommen haben. Wo man eben noch im Detail definiert und programmiert. Heute werden vor allem bestehende Routinen und Programme gesucht und dann eingebunden in ein größeres Ganzes. Die Sprachen, die man dazu verwendet sind schon ein bisschen anders. Man nennt sie Scripting-Languages. Ich finde das schade. Mich interessiert das weniger als das originelle Programmieren mit mathematischen Prinzipien. Heute sucht man einfach et-

was, das mehr oder weniger in das Puzzle passt und das Resultat ist, dass die Programme viel größer werden als eigentlich notwendig, da diese Puzzleteile immer auch bisschen etwas Anderes tun, das man gar nicht braucht. Das ist aber der Trend und der wird gefördert durch diese riesigen Computerleistungen und riesigen Speicher. Also irgendwie ökonomisieren, das lohnt sich ja kaum mehr, das ist schade. Das ist doch eigentlich das Grundprinzip des Ingenieurs, dass er versucht, aus möglichst wenig möglichst viel zu herauszuholen.

Gutknecht: Etwas, das mich immer gestört hat, wenn du von Fahnenstange sprichst: Man hat immer noch zwei Welten. Auf der einen Seite die Objekte, die irgendwelche Zustände beschreiben und auf der anderen Seite etwas wie Threads oder Prozesse, die diese Objekte steuern. Ich würde gerne sehen, dass man die beiden Konzepte vereinigt. Es geht bei der objektorientierten Programmierung ja vor allem darum, dass man möglichst irgendwie die echte Welt abbildet und in der echten Welt haben alle Objektebestimmte Zustände. Meiner Meinung nach wäre das Ende der Fahnenstange, wenn die beiden vereinigt werden, sozusagen wie active Objects, wenn die beiden Welten der Prozesse oder Threads einerseits und der Objekte vereinheitlicht wären.

Wirth: In der Praxis redet man nicht mehr

groß über Objektorientierung.

Gutknecht: Du hast schon richtig gesagt, es geht in eine etwas andere Richtung mit diesen Scripting-Sprachen und mit diesen reusable Systems etc., aber das heißt nicht, dass aus Sicht der Lehre das Ende der Fahnenstange erreicht ist.

Wirth: Python ist ja extra so und das gefällt mir.

Gutknecht: Wenn wir schon von Python reden: es gibt ein extrem hässliches Konzept und das ist, dass man die Positionierung innerhalb von Spalten zur Syntax nimmt. Das heißt, eine Einrückung hat eine syntaktische Bedeutung. Ich finde, für die Schulung ist das extrem hässlich.

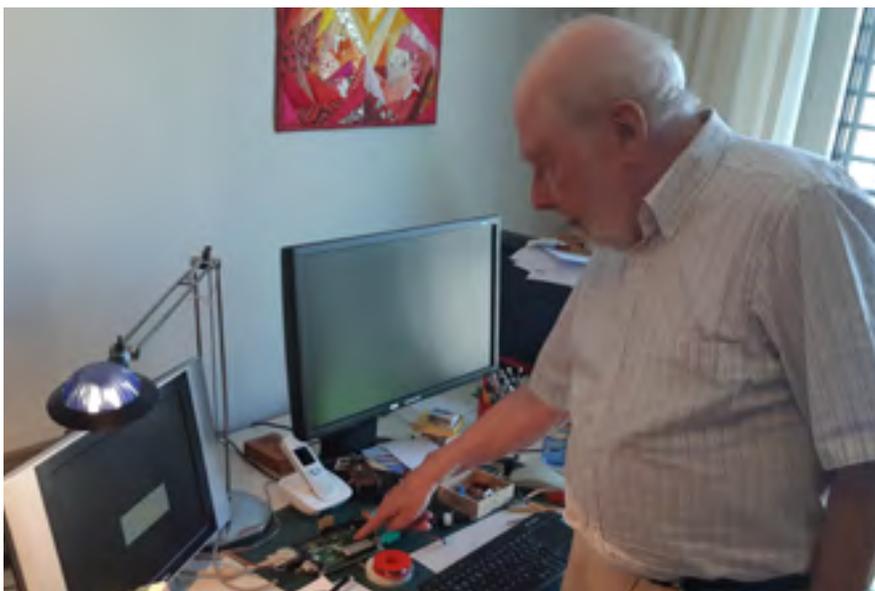
Wirth: Ja, aber da darf ich vielleicht sagen, von Pascal aufwärts habe ich immer die Schulung sehr im Vordergrund gesehen. Ich wollte nie irgendein Gegenstück zu kommerziellen Riesenprojekten vorschlagen, sondern etwas, dass die Schulung fördert, die richtigen Grundlagen richtig vorzeigt.

Gutknecht: Da muss ich auch noch kurz was sagen, das hat mich immer beeindruckt. Als ich in die ETH gekommen bin, hat mich immer diese saubere Trennung zwischen Syntax und Semantik und Darstellungssyntax und Semantik beeindruckt. Das waren für mich wirklich wichtige und verschiedene Konzepte und bei Python wird alles rausgeworfen. Das ist ausbildungstechnisch höchst schädlich.

Wirth: Absolut. Dennoch wird es hochgejubelt, weil es irgendwie praktisch ist oder einfach, weil es populär ist.

Sie haben kurz angesprochen, dass die Scripting-Languages immer mehr ein großes Ganzes werden. Die Wirtschaft muss derzeit mit immer größer werdenden Systemen leben, die kaum mehr beherrschbar sind. Was sind die Ursachen für die Misere? Hat man zu wenig auf Sie gehört? Wie würden Sie diese Entwicklung auch im Bereich Sicherheit sehen? Für Sie war ja immer die Rückbesinnung auf einfache Algorithmen wichtig.

Wirth: Ja, das kommt ganz einfach aus meiner eigenen Beschränktheit. Wenn es zu kompliziert wird, kann ich den Überblick nicht mehr halten und das Einzige, wie ich komplizierte Aufgaben lösen kann, ist durch einfache Lösungen. Die



Einfachheit, die Struktur, die Regularität, das ist für mich immer die Lösung. Dazu gehört auch ein Werkzeug, eine Sprache, die das unterstützt.

Die Programmiersprache Oberon wird auch in Kernkraftwerken verwendet. Man überlegt gerade, sie auch in Smart-Energy und Smart-Meter einzubauen, weil Oberon sehr sicher und sehr einfach aufgebaut ist. Wäre der von Ihnen von Grund auf minimalistisch konzipierte Computer eigentlich zu hacken? Hätte die NSA eine Chance?

Wirth: Ich denke schon. Kernkraftwerk habe ich auch einmal gehört, aus Russland.

Gutknecht: Ja genau, aus Russland. Ich war das letzte Mal am Oberon-Tag in Russland und da wurde mir sehr klar gesagt, dass dort für die Steuerung von Atomkraftwerken eigentlich nur Oberon in Frage kommt.

Wirth: Da muss ich vielleicht noch einfügen: Höhere Programmiersprachen definieren ja eine höhere Ebene der Abstraktion und dann gibt es einen Compiler, der diese Konzepte, diese Statements in Sequenzen von Computerbefehlen übersetzt. Da gibt es einen Gap zwischen der höheren Sprache und der niederen Computerbefehlsfolge. Es war immer mein Ziel diesen Gap klein zu halten. Es sollte eigentlich immer sogar für einen Laien möglich sein, aus einer Folge von Programmieranweisungen ungefähr herauszufinden wie der Computer das bewältigt, also welche Befehlsfolgen daraus gemacht werden. Bei diesen Monstersprachen ist das absolut unmöglich. Sie sind viel zu komplex und deshalb sind auch diese Compiler so furchtbar riesig, kompliziert und langsam. Etwas, das man zu Oberon noch erwähnen sollte: Die Übersetzung geht rasant! Das ist

eine Zehntel-Sekunde für ein Mehrseiten-Programm. Das spielt heute keine Rolle mehr. Die Computerleistungen sind ja fast unendlich.

Aber ich glaube, da wird es wieder einen Trend in die andere Richtung geben. Lassen sich komplexe kommerzielle Systeme, wie Windows oder IOS überhaupt 100%ig sicher bauen?

Wirth: Ich glaube nicht. Ich weiß nicht, wie man da wirklich die Übersicht behalten kann mit derart komplexen Systemen. Aber es gibt keinen Weg zurück. Das ist das Verrückte. Und wollten Sie jetzt Oberon als Idealsystem anerkennen, gäbe es keinen Weg, das in der Welt heute noch einzuführen. Die ganze Welt ist total abhängig geworden von diesen industriellen Produkten. Schade. Ausser vielleicht in einzelnen Gebieten, wo es um Leben und Tod geht. Was vor allem wünschenswert wäre: In Schulen, in Universitäten, darauf hinzuweisen, denn dort werden ja Grundlagen gelehrt. Das hat sich bisher nach meiner Ansicht nicht durchgesetzt.

Sie hatten sich in der ETH-Zeit, wo Sie von 1968 bis zu Ihrer Emeritierung 1999 gelehrt haben, immer mehr auf FPGAs spezialisiert und konzertiert, wo Sie, von der Elektrotechnik kommend, Software mit Hardwareingenieuren konnten. Können Sie dazu etwas erzählen?

Wirth: Die Idee, Hardware als Interessensgebiet wieder aufzugreifen hatte ich während eines Sabbaticals [bei Xerox]. Dort hatte ich den ersten Personal Computer zur Verfügung, für mich alleine. Das war kein Spielzeug, sondern ein echter Computer, wo man Compiler bauen konnte. Es war mir völlig bewusst, dass ich nach diesem Jahr nicht mehr nach Zürich heimkehren kann, um wieder mit diesem Monstrum im Keller zu arbeiten. Interaktivität war offensichtlich die Sa-



Lilith mit Maus (Sammlung der ETH Zürich)

che der Zukunft. Nur konnte man diesen Computer nicht kaufen, das war eine interne Sache, also blieb nur der Entschluss, selbst so etwas zu bauen. Da habe ich mich dann intensiv mit Hardware abgegeben und den Computer Lilith gebaut, mehr oder weniger nach dem Vorbild dieses Alto by Xerox.

Den habe ich gerade an der ETH Zürich gesehen, da steht ein Exemplar vom Lilith, mit einer kleinen Computermaus. Ich habe da erfahren, dass Sie diese Computermaus auch von Xerox mitgebracht haben und Logitech hat das dann übernommen.

Wirth: Ja, das stimmt. Das war ein wichtiger Bestandteil dieses Alto und später der Lilith und das gab es damals auch nicht zu kaufen. Ich habe dann einen Kollegen gefunden, der gesagt hat, dass es ihm möglich wäre so etwas zu bauen. Er hatte einen Uhrmacher in seinem Labor und der hat dann sowas gebaut. Die Logitech-Maus wurde später daraus. Aber Logitech gab es, soviel ich weiß, damals noch nicht. Das war wahrscheinlich um 1978.

Fortsetzung in der nächsten Ausgabe des OCG Journals

Zemanek
Lectures

Erleben Sie Niklaus Wirth live!

Am 1. Jänner 2020 wäre der österreichische Computerpionier Prof. Heinz Zemanek 100 Jahre alt geworden. Dieses Jubiläum nimmt die OCG zum Anlass, die **Zemanek Lectures** zu starten.

Die **1st Zemanek Lecture** wird am **14. Jänner 2020** im Kuppelsaal der TU Wien stattfinden. Als Referenten konnten wir Niklaus Wirth gewinnen der über *50 Jahr Pascal* referieren wird.

Weitere Informationen: www.ocg.at/100-jahre-heinz-zemanek



Computerpionier Heinz Zemanek wird mit einer Vortragsreihe geehrt

von Katharina Resch-Schobel

Die erste Zemanek Lecture

Bei der Auftaktveranstaltung der Zemanek Lectures der OCG im voll besetzten Kuppelsaal der TU Wien am 14. Jänner teilten Wegbegleiter des Visionärs Heinz Zemanek mit einem Expert*innenpublikum ihre Erfahrungen. Mit dem Turing Award Preisträger Niklaus Wirth als ersten Sprecher der in Zukunft jährlich stattfindenden Zemanek Lectures wurde die Latte sehr hoch gelegt.

Die „Entwicklungen rund um die Digitalisierung machen manchmal Angst, daher müssen diese Entwicklungen von allen Seiten kritisch betrachtet werden – hier agiert die Österreichische Computer Gesellschaft seit ihrer Gründung 1975 als Think-Tank“, sagt OCG Präsident Wilfried Seyruck in seinen Eröffnungsworten. So zieht sich dann auch durch die ganze Veranstaltung der Leitgedanke, dass bei allem Fortschritt immer der Mensch im Mittelpunkt steht und die Möglichkeiten der Informatik als Herausforderung für den Menschen und die Gesellschaft zu betrachten sind.

Im Bundesministerium für Verkehr Innovation und Technologie (BMVIT) hat die OCG von Anfang an einen wichtigen Unterstützer bei der Umsetzung ihrer Agenden gefunden. Michael Wiesmüller begrüßt die Gäste im Namen des BMVIT, freut sich über den „mit IKT-Kompetenz gefüllten Raum“ und verspricht auch weiterhin seine Verbundenheit mit der OCG und ihren Zielen.

In teilweise sehr persönlichen Statements schließen Gerti Kappel (Dekanin TU Wien), Peter Reichel (Österreichischer Verband für Elektrotechnik OVE), Erich Neuhold (International Federation for Information Processing IFIP) und Peter Reichl (Universität Wien) an die Visionen Zemaneks an und drücken ihre Wertschätzung für den Computerpionier, der „überzeugen und begeistern konnte“, aus.

DAS BLEIBENDE ABER IST DER MENSCH

„Die Informatik ist eine politische Disziplin

geworden“, ist Peter Reichl (Universität Wien) überzeugt und erinnert an Heinz Zemaneks Schlussworte bei einem seiner Vorträge: „Das Bleibende aber ist der Mensch!“. Bernhard Schmid von der Wirtschaftsagentur Wien freut sich, dass die Veranstaltungsreihe Zemanek Lectures 2020 zum Ziel der Stadt Wien beiträgt, Digitalisierungshauptstadt zu werden. Peter Lieber (VÖSI) wünscht sich für Österreich manchmal mehr Mut und Tempo, wenn es um Innovation geht – und ist zuversichtlich, dass die neue Regierung das Thema Digitalisierung vorantreiben wird.

Auch Norbert Rozsenich und Gerhard Chroust stellen in ihrer Laudatio an Heinz Zemanek den visionären Menschen und Wissenschaftler in den Vordergrund, der sich immer auch philosophischen Fragen gestellt hat. Transdisziplinäre Forschung, internationale Vernetzung und leidenschaftliche Überzeugungskraft zeichneten das Wirken von Heinz Zemanek aus, der bis an sein Lebensende ein hochge-

schätzter Vortragender war. Im Kern sah er sich immer auch als Ingenieur, daher auch sein Ausspruch „Wahr ist, was funktioniert“. Mit seiner Begabung auch Laien komplexe Inhalte verständlich zu vermitteln ist ein Vorbild für WissenschaftlerInnen alle Disziplinen.

Niklaus Wirth spannt in seinem Vortrag 50 Jahre Pascal einen Bogen von den 1960er Jahren bis in die Gegenwart. Sein ganzes Forschungsleben plädierte er für

einfache Algorithmen und schlanke Software. „Software wird schneller langsamer, als die Hardware schneller wird“ lautet das Wirthsche Gesetz. Heute mahnt er, nicht nur Moderne Trends und anwendungsorientierte Themen an den Universitäten zu lehren, sondern grundsätzliches Verständnis zu fördern und damit Wissen, das auch noch 20 Jahre später Gültigkeit hat.

Franz Zeller, ORF, führte fachkundig

durch den Abend und es gelang ihm auf gewohnt wertschätzende Art mit den Gästen ins Gespräch zu kommen.

Die OCG führt das Werk Heinz Zemaneks weiter – sie will Menschen für die IKT begeistern, als beratender Think-Tank für die politischen Entscheidungsträger*innen, mit einem kritischen Blick auf die Technik und immer mit dem Menschen im Mittelpunkt.



Familie Zemanek mit den Vortragenden
Alle Fotos: Barbara Lachner | Photosandmore.at



Der Kuppelsaal der TU Wien war der perfekte Rahmen für das Event.



Prof. Gerti Kappel, Dekanin an der Informatikfakultät der TU Wien



OCG Präsident Wilfried Seyruck



Unterstützer Zemanek Lecture 2020

Wir danken unseren Organisationspartnern für die erste Zemanek Lecture: BMK, Universität Wien, OVE, IFIP, TU Wien, VÖSI sowie Wirtschaftsgesellschaft Wien

Zemanek Lecture 2021

Termin: **1. Juli 2021**

Vortragender:
Douglas R. Hofstadter, Physiker, Informatiker und Kognitionswissenschaftler

von Ronald Bieber

Streben nach Wissen: Pascal, Modula und Oberon

Lesen Sie auf den folgenden Seiten das Gespräch, das am 16. September 2019 in Zürich stattfand.

Ich habe gehört, dass einige Ihrer Schüler in die Welt ausgeschwirrt sind und einiges bewegt haben. Jürg, du bist auch ein Schüler von Professor Wirth. Einer davon ist Urs Hölzle, der nach Amerika zu Google gegangen ist.

Wirth: Hölzle hat bei Jürg Gutknecht diplomiert und ist dann nach Stanford gegangen. Er hat sicher damit zu tun, dass Google so gut vertreten ist in Zürich

Gutknecht: Es ist damals schon aufgefallen, dass er jemand Spezieller ist und Initiative hat.

Wirth: Ein sehr aktiver Kerl, Verbindung von Wissenschaftlichem und Kommerziellem.

Gutknecht: Er hat sich damals speziell interessiert für die Programmiersprache Self und er wollte sich eigentlich in die Programmiersprachen-Branche weiterentwickeln und hat dann offenbar Leute in Google gefunden.

Wirth: Darf ich noch einen kleinen Nachtrag geben zur Sache der Programmiersprache und wie sie gelehrt werden? Ein Kollege hat schon etliche Male eine Vorlesung in Hongkong gehalten. Da hat er letztes Mal gesagt, er hätte eine Aufgabe an die Studenten ausgegeben: „Programmieren Sie etwas, was das und das und das vollbringt.“ Danach sind alle aufgestanden und hätten sich sofort ihren Handys und Computern zugewendet, um im Internet zu suchen, ob so etwas nicht schon bestehe. Das eben als

Nachtrag zu diesem Konzept, dass man Bausteine sucht, die man dann mehr oder weniger gut zusammenfügt. Das ist natürlich dann nicht mehr das Lernen von Konzepten und von Konstruktionsmethoden. Das ist Internetsuche und „Zusammenklätschen“. Das ist der Trend heutzutage.

Wobei das allein nicht genügen wird. Wie müssen Ihrer Meinung nach Systeme entworfen werden, um einfach und zuverlässig zu funktionieren?

Wirth: Wenn ich das in ein paar Worten ausdrücken könnte, wäre ich sehr glücklich. Ich weiß es nicht, aber dazu gehört ganz sicher eine Menge Erfahrung. Das können Sie natürlich nicht in ein oder zwei Vorlesungen an der Universität erwerben. Das muss sich über die Jahre anhäufen. Selbst wenn man alt ist, merkt man immer noch, dass man verbessern kann. [Edsger] Dijkstra hat einmal gesagt: „Das Vermeiden von unnötiger, hausgemachter Komplexität ist das Wichtigste, das ein Programmierer jeden Tag üben muss.“ Die Probleme, die uns vorgelegt werden, sind kompliziert genug, man muss sie nicht noch künstlich weiterkomplizieren.

Damit bin ich schon bei meiner nächsten Frage: Was macht einen guten Softwareentwickler Ihrer Meinung nach aus?

Wirth: Ich glaube schon, dass es das ständige Überdenken ist. Könnte man das nicht einfacher machen und meistens damit auch noch besser lösen? Aber Programmierer sind überall unter hohem Zeitdruck und der verunmöglicht das. Es lohnt sich nicht darüber nachzudenken.

Der Computer muss ein wenig schneller sein und dann merkt niemand die Komplexität. Ich glaube aber schon, dass die Suche nach guten und einfachen Lösungen, überzeugenden Lösungen, das Wichtigste ist.

Im deutschsprachigen Raum haben wir das Problem, dass in der Informatik Frauen unterrepräsentiert sind. Es beginnt im Studium und geht bis zur Arbeitswelt. Was haben Sie für Empfehlungen, um da eine Trendwende zu schaffen? Wir sprechen immer vom Nachwuchskräftemangel.

Wirth: Ich glaube die Trendwende ist schon eingetreten. Zu meiner Zeit gab es an der ETH keine Mädchen und das hat sich schon geändert. Dazu braucht es einfach Zeit und natürlich gibt es Programme, und das ist auch gut so, um Frauen etwas zu animieren in technischen Berufen mitzumachen. Ich glaube Informatik ist prädestiniert dazu, dass Frauen im gleichen Anteil teilnehmen. Ich bin gegen künstliche Programme und Quoten und so. Ich glaube aber, das wird sich schon einpendeln. Es stimmt aber, gerade in der Schweiz sind bei Frauen technische Berufe immer noch relativ unbeliebt. Das ist einfach so. Sie können ja Ihrer Tochter lange erklären, dass sie jetzt Mathematik studieren muss. Wenn sie nicht will, dann geht es nicht.

Gutknecht: Ich dachte, gerade in der Mathematik ist die Geschlechterpräsenz ausgeglichen, oder nicht? Eigentlich sollte das in der Informatik ebenso sein.

Wirth: Das ist auch in vielen Orten so, z. B. in der USA und in China, aber nicht in der Schweiz.



Niklaus Wirth bei der ersten Zemanek Lecture. Mit im Bild: Ronald Bieber, Generalsekretär der OCG und Wilfried Seyruck, Präsident der OCG. Alle Fotos: Barbara Lachner | Photosandmore.at

Erst in den letzten Jahren erfolgte der Durchbruch von Machine-Learning, das inzwischen mit künstlicher Intelligenz gleichgesetzt wird. Wie haben Sie diese Entwicklung, wie z. B. die Triumphe von AlphaGo erlebt?

Wirth: Das ist ein Thema und ich bin ziemlich weit herum bekannt als Skeptiker von Artificial Intelligence, aber ich darf Ihnen gleich erklären warum. Ich habe schon 1964 mit Exponenten von künstlicher Intelligenz gesprochen, wie [John] McCarthy, [Joseph] Weizenbaum und [Edward] Feigenbaum, und ich habe dann immer empfunden, es sei etwas Großtuerei, dass man riesige Versprechen macht, die dann doch nicht eingelöst werden, um von den Funding Agencies Geld zu bekommen. Tatsächlich sind die Versprechungen immer erst viel später, aber doch immerhin, eingelöst worden. Heute hat sich das Blatt gewendet. Man kriegt den Eindruck, dass Computer Science fast gleich steht mit künstlicher Intelligenz. Das ist natürlich jetzt, weil das Wort auch unter die Politiker und unter das Volk geraten ist und man sich gerne Wunder verspricht. Aber Wunder sind nicht zu erwarten. Auf der anderen

Seite muss man anerkennen, dass auch Fortschritte erzielt wurden, also Machine-Learning ist ein Beispiel dafür. Aber wie gesagt, Wunder würde ich nicht erwarten.

Eine Kurzbeschreibung von Machine-Learning lautet ja: Daten schreiben Programme. Lässt sich Programmieren automatisieren oder wären Programmierer in der Zukunft wegrationalisierbar?

Wirth: Nein, das glaube ich sicher nicht. Machine-Learning heißt, dass man aus Daten, Beispielen lernt, wie die Zusammenhänge sind und dann auch das Programm anpasst. Meistens wird gar kein Programm geändert, sondern Bedingungen werden anders eingestuft. Das Schlagwort nach meiner Meinung ist Heuristik. Man bezieht vergangene Erfahrungen mit in die Verarbeitungsmethodik der Zukunft ein. Wo ich immer noch skeptisch bin ist dort, wo Programme verwendet werden, deren Resultate nicht voraussehbar sind, also nicht deterministische Programme, und wo sich die Resultate auch nicht wiederholen lassen, weil man Erfahrungen in einer nicht übersehbaren Art und Weise bezieht. Ich habe den Computer immer als Maschine aufgefasst, und wenn man mir genü-

gend Daten gegeben hat, dann kann ich voraussagen, was das Resultat sein wird. Das ist in der heutigen Zeit nicht mehr üblich, also mehr und mehr vertraut man eben auf Heuristiken und schaut dann einfach, was das Resultat ist. Das ist vielleicht nicht so tragisch, aber es gibt Gebiete, wo man das vermeiden sollte. Denken Sie an medizinische oder militärische Anwendungen.

Wie schätzen Sie die aktuellen Entwicklungen auf dem Gebiet der Rechnerarchitekturen, der Programmiersprachen und der Softwaretechnik ein? Was davon brennt Ihnen am Ehesten am Herzen?

Wirth: Man redet auch immer wieder von neuen Computerarchitekturen. Ich habe eigentlich noch keine grundlegenden Änderungen wahrgenommen. Natürlich werden die Computersysteme immer komplexer, und zwar zum Zweck der Optimierung. Die verrücktesten Dinge werden unternommen, um ein paar Instruktionen sparen zu können.

Neben Artificial Intelligence ist ja Quantencomputing das zweite große Schlagwort heutzutage. Ich verstehe Quantencomputer nicht, weil ich Quantenphysik kaum verstehe. Ich muss ein-

fach zur Kenntnis nehmen, was darüber berichtet wird, aber ich glaube, es handelt sich um eine völlig andere Art von Computer, als wir es heute gewohnt sind und ich glaube, man sollte nicht allzu viele verfrühte Hoffnungen in diese Dinge stecken.

Gibt es jetzt aus Ihrer Sicht Fehlentwicklungen im Bereich der Informatik bzw. Dinge, wo Sie sagen: Das passt absolut, das geht in die richtige Richtung?

Wirth: Heute geht ja alles in alle Richtungen.

Was wäre Ihr Wunsch an die Zukunft? Wie soll sich die Informatik weiterentwickeln?

Wirth: Gerade in der Ausbildung: Dass man die Grundprinzipien, Grundkonzepte in den Vordergrund stellt, und dass dann die selbstdenkenden Menschen ableiten können, was man mit diesen Grundprinzipien alles machen kann und nicht einfach Rezepte verteilt, ganz ähnlich wie wir vorher bemerkt haben mit diesen Internetsuchen.

In der Ausbildung ist Computational Thinking ist ein wichtiger Ansatz.

Wirth: Ja genau, jawohl.

Ich glaube die Schweiz hat das erfolgreich in den Lehrplan 21 umgesetzt. Die OCG setzt sich sehr dafür ein, Kinder und Jugendliche für Informatik früh zu be-

geistern. Wir haben den Biber der Informatik Wettbewerb. Da geht es genau in dieses Computational Thinking hinein, wo letztes Jahr 32.000 Schülerinnen und Schüler von 8-18 Jahren mitgemacht haben. Wir versuchen, auch in Computing-Kursen, Kindern algorithmisches Denken mitzugeben. Wie stehen Sie zu solchen Initiativen?

Wirth: Die habe ich eigentlich immer als positiv empfunden.

Haben Sie Empfehlungen, die Sie der SI und der OCG mitgeben können? Worauf sollen wir noch mehr, auch in der Politik, achten?

Wirth: Das ist so eine Sache. Man darf es ja auch nicht übertreiben. Man kann diese jungen Leute nicht zwingen nur noch an Informatik zu denken und nur noch mit dem Handy herumzuwandern, aber richtig eingesetzt ist es sehr wichtig; z. B. in normalen Schulfächern wie Mathematik, Physik, da ist es immer gut, wenn man die Informatik miteinbezieht, wo es sinnvoll ist. Es gibt ja kaum Gebiete, wo der Computer keine Rolle mehr spielt. Vielleicht Theologie.

Selbst die Kirche würde keine Kirchensteuer mehr bekommen, hätte sie keine guten Computersysteme. Welche Informatikkenntnisse sollten heute bereits an den Schulen gelehrt werden?

Wirth: Ich bin immer ein konservativer

Mensch gewesen diesbezüglich. Dass man schon in der Unterstufe Informatikprinzipien lehren muss, ist nach meiner Ansicht nicht unbedingt nötig. Wo eben ein analytisches und mathematisch fundiertes Denken möglich wird, da gehört es sicher hin. Wenn man „Informatik“ sagt, muss man gleich fragen, was damit gemeint ist. In den schweizerischen Schulen meinen sie meistens die Anwendung von Handys für Text-Editing und Massiv-Sending und vielleicht PowerPoint, also Anwendungen. Das ist nicht, was ich unter Informatik verstehe. Das gehört meiner Meinung nach eher ins Gymnasium, wo es dann wirklich um Konzepte der Informatik geht, nicht nur um Anwendungen.

Wir als OCG stehen ja auch hinter dem Europäischen Computer Führerschein, der genau auf diese Anwendungskennnisse hinzielt.

Wirth: Das ist auch wichtig, ich möchte das überhaupt nicht kleinreden. Wie man früher Schreibmaschinenstunden nehmen musste, muss man heute Computerstunden nehmen.

Sie hatten in Ihrer aktiven Zeit Kontakten mit österreichischen Forschern, z.B. an der JKU Linz. Haben Sie auch im Ruhestand noch Kontakte zu diesen Instituten?

Wirth: Nur auf privater Ebene. [Gustav] Pomberger kenne ich noch und [Hanspe-

Niklaus Wirth im Gespräch mit Ronald Bieber (OCG) und Michael Wiesmüller (BMK)

Peter Reichl, Universität Wien, organisierte im Rahmen der Lecture eine kleine Hardwareausstellung.



ter] Mössenböck natürlich und [Laszlo] Böszörményi in Klagenfurt. Mit denen habe ich am ehesten noch Kontakte. Sonst habe ich mich ziemlich zurückgezogen. Ein Projekt darf ich jetzt vielleicht doch noch erwähnen.

Woran Sie gerade arbeiten?

Wirth: Auch das ist jetzt bereits am Ausklingen. Wir [Wirth und Gutknecht] haben ja zusammen ein Buch geschrieben, 1992. Das nennt sich Project Oberon und da haben wir unser ganzes Projekt von A bis Z beschrieben, in einem einzigen Buch. Die Entwicklung des Betriebssystems, des Compilers und von gewissen Anwendungen, Grafik-Editor Oberon. Das Buch ist sehr detailliert, ist sehr technisch und hat deshalb relativ wenig Anklang gefunden. Dann ist ungefähr zwischen 2000 und 2005 ein Engländer zu mir gekommen und hat gesagt, er hätte aus diesem Buch sehr viel gelernt. Endlich ein Buch, wo man sagt, wie man etwas tut und nicht nur was. Es wäre doch schön, wenn ich dieses Buch neu beleben könnte mit einer neuen Ausgabe. Zuerst habe ich gezögert, aber dann doch zuge sagt. Da war dann das Kapitel über den Compiler. Das war natürlich ein Unikum, denn der Computer, für den kompiliert wird, den gibt es nicht mehr, schon lange nicht mehr. Also was machen? Ich muss es sowieso auslassen oder neu schreiben. Dann hat er gesagt, dass ich dann eben meinen eigenen hypothetischen Computer konzipieren oder bauen soll. Das kann man ja simulieren. Dann ist dieser Paul Reed gekommen und hat gesagt: Ja, jetzt gibt es diese programmierbaren Bausteine, also FPGAs, Field Programmable Gate Arrays. Da könntest du ja diesen Computer selber bauen. Das ist ja nur ein Hardwareprogramm, das du schreiben musst. Das habe ich dann in einer kurzen Zeit gemacht und dieses Ding habe ich dann zu Ende geführt und sogar das Kapitel neugeschrieben. Ich kann Ihnen das nachher selber schnell zeigen. Das ist jetzt mein letztes Projekt, das ich mehr oder weniger abgeschlossen habe. Tatsächlich ist dieses FPGA im Prinzip eigentlich sehr ineffizient. Aber dieser Computer läuft so schnell wie der Computer, den ich 1990 gebaut habe für Oberon. Erstaunlich.



Gerhard Chroust und Norbert Rozsenich erinnerten an Heinz Zemanek, hier im Bild mit Niklaus Wirth und Ronald Bieber, OCG.

Gibt es von dem Buch schon eine fertige Ausgabe?

Wirth: Nein, das Buch nicht, aber es ist alles im Internet frei verfügbar. Das Ganze heißt: Project Oberon. Es ist unter meiner Webseite und auch unter ProjectOberon.com zu finden.

Die OCG ist ja auch ein Verlag. Wir könnten das mit Ihnen publizieren, wenn Sie wollen.

Wirth: Können Sie gerne, ja.

Welche Empfehlung oder Ratschläge können Sie Computergesellschaften, wie der SI oder der GI in Deutschland, die heute den 50er feiert, oder der OCG, die nächstes Jahr den 100sten Geburtstag ihres Gründers, Heinz Zemanek feiert, mitgeben?

Wirth: Ich weiß nicht, ob ich da einen Ratschlag mitgeben kann, aber ich kann Ihnen sagen, was mir etwas Bauchweh macht in der heutigen allgemeinen Entwicklung und das ist, dass alles nur noch nach Geld orientiert ist. Man lernt etwas, um sein Know-how zu erweitern und um in neue Gebiete, in denen Geld eine Rolle spielt, einzutreten. Ich glaube ich darf sagen, zu meinen Zeiten hat man noch mehr studiert, einfach um der Neugier Willen. Man wollte etwas erlernen, Wissen nähren. Natürlich brauchte man das später auch zum Erwerb, aber es stand nicht so sehr im Vordergrund. Also wenn ich auf das Geld geschaut hätte, wie das heute üblich ist, wäre ich nicht Professor

geworden. Ich habe mit Pascal und Oberon nichts verdient. Ich war ein guter Lehrer an der ETH und das hat mit genügt. Aber heute ist schon das Streben nach Geld dominant. Ob das Computergesellschaften etwas nützen kann, wenn ich das sage, weiß ich nicht.

Sie haben drei Kinder, zwei Töchter und einen Sohn. Ist jemand in die Richtung der Informatik gegangen?

Wirth: Nein, keiner. Schlimmer noch: Sie haben das immer möglichst vermieden. Aber meine Tochter war die ältere, die ist heute 100 Mal gerissener in der Computerbenützung als ich. Das heißt natürlich Internet Search. Sie hat ein kleines Lebensmittelgeschäft und da geht heute nichts mehr ohne das Handy. Von Bestellungen zu Bezahlungen ist alles möglich, alles. Da ist sie wirklich gerissen.

Hier in Zürich?

Wirth: Ja, ganz in der Nähe.

Sehr schön. So schließt sich dann der Kreis.

Wirth: Der Sohn, der ist Musiklehrer, Musiker. Damit habe ich wiederum nichts zu tun gehabt.

Zur Verknüpfung von Informatik und Musik wird aber intensiv geforscht. Ich denke da etwa an die Arbeit von Gerhard Widmer an der JKU Linz.

Wirth: Ja, Musik und Mathematik werden ja oft verknüpft.

Herzlichen Dank für das Interview.