

Im OUTPUT-Gespräch: Prof. C. A. Zehnder:

Wege der Informatikausbildung

Interview: Louis A. Venetz

Professor Dr. Carl August Zehnder ist ursprünglich Mathematiker, hat aber seit über 20 Jahren einen Lehrstuhl für Informatik an der ETH Zürich. Er ist ein namhafter Pionier der Schweizer Informatikszene und Autor von einigen Publikationen und Lehrbüchern über Datenbanken und Informatik-Projektführung. Von 1987 bis 1990 war er vollamtliches Mitglied der Schulleitung der ETH Zürich. Jetzt ist er zurück in der Informatik und hat eben eine neue Vorlesung gehalten: «Gestaltung grosser Informationssysteme». Er wirbt dabei für föderalistische Lösungen sowie allgemein für eine gesunde Entwicklung der Informatikausbildung in der Schweiz.

OP: Herr Prof. Zehnder, wie ist es Ihnen in Verlauf der Jahre gelungen bei dieser Informationsflut den Überblick zu behalten, die Informationen aufzubereiten und sie an die Studenten weiterzugeben?

Prof. Zehnder: Ich habe immer wieder Gelegenheiten ergriffen, an Aufgaben aus der Praxis mitzuarbeiten, z.B. als Berater bei Evaluationen, Projektleiter, Kommissionsmitglied und zuletzt als Mitglied der ETH-Schulleitung – die ETH Zürich ist schliesslich ein Grossbetrieb, der jährlich um die 50 Mio Franken für die Beschaffung von Informatikmitteln extern ausgibt. Bei einer solchen Mitwirkung geht es aber für einen Hochschullehrer nicht einfach darum, rasche Lösungen zu finden, sondern auch Grundsatzfragen aufzuwerfen und zu beantworten. Eine Hochschule ist nicht bloss eine Lehranstalt; hier ist die Lehre immer gepaart mit «selber tun». Die Dozenten eines Ingenieurgebietes können ihr Wissen nicht nur über die Literatur

aufnehmen, sondern müssen auf ihrem Lehrgebiet auch in Forschung und Entwicklung mitarbeiten und mit Anwendungen zu tun haben.

OP: Hat die ETH Zürich die Informatik nicht erst viel zu spät entdeckt? Sie bietet doch erst seit 1981 ein Informatikstudium an?

Prof. Zehnder: Ich habe schon 1958 im 3. Semester an der ETH Zürich programmieren gelernt, denn die ETH hat schon sehr früh diesbezüglich Fakultativausbildungen angeboten. Es brauchte aber damals einen zusätzlichen Effort, damit man überhaupt in die Informatik hineinkam. Seit 10 Jahren haben wir nun eine Abteilung für Informatik, diese ist z.Zt. die drittgrösste Abtei-

lung an der ETH mit etwa 1000 Studenten.

OP: Ich habe noch eine Frage zum Schulsystem an der ETH. Im alten Griechenland hat Sokrates die Schüler durch Fragen zum Selberdenken angeregt. Heute wird kursmässig doziert. Warum?

Prof. Zehnder: Das heutige Schulsystem ist primär eine Frage von Kosten und Nutzen. Es gibt viele Hochschulen mit einem relativ starken und klaren Kurskonzept. Die ETH Zürich beispielsweise kann stolz darauf sein, dass sie zu denjenigen europäischen Hochschulen zählt, welche die kürzeste Studiendauer haben. Die Studenten brauchen rund 9 Semester bis zum Diplom, und bei durchschnittlich 9,4



Wir müssen den Einstieg in die akademische Berufswelt so gestalten, dass die erste Schulung relativ kompakt ist und nachher ein Ausbildungsrecycling stattfindet.

Semestern sind sie fertig. Gegenüber «ewigen» Studenten ist das natürlich schon ein Pluspunkt. Es genügt ja nicht, dass jemand ein Studium macht – auch wenn es noch so lang und umfangreich ist – und nachher meint, er könne 30 Jahre lang davon profitieren. Das geht nicht mehr! *Wir müssen den Einstieg in die akademische Berufswelt so gestalten, dass die erste Schulung relativ kompakt ist und nachher ein Ausbildungsrecycling stattfindet.* Ich meine keineswegs, dass jeder Informatiker unbedingt durch die Hochschule gegangen sein muss. Für mich sind bei einem 40jährigen Informatiker in der Praxis der Weg über die Hochschule, der Weg über die HTL und der Weg über die Praxis mit höheren Berufsprüfungen weitgehend äquivalent für die Anwendung. Selbstverständlich ist es für die Forschung etwas anders. Die Hochschule hat aber in unserer Gesellschaft die Aufgabe, dass sie eine *grössere Gruppe* von Leuten zu diesen Fähigkeiten führt, zügig, effizient. Es gibt auch Genies in unserer Gesellschaft, aber für die müssen wir nicht besonders Sorge tragen, die machen ihren Weg sowieso. Und Leute, die aus irgendwelchen Gründen besondere Wege gehen, darf man deswegen nicht strafen. Eine vernünftige Wirtschaft gibt allen ihren Platz. Aber es kann nicht Aufgabe der Hochschule sein, dieser Spezialfälle wegen ihre Hauptaufgabe zu vergessen, und das ist der durchschnittliche Student. Und bei diesem haben wir an der ETH einen recht hohen Standard. Entsprechend hoch sind auch die Durchfallquoten, in der Informatik sogar etwas allzu hoch. Aber das hat seinen Grund. Im Informatikstudium gibt es nämlich auch einzelne Studenten, die gar nicht hierher gehören, die von der Mittelschule her den Eindruck haben, dass man es mit «Töggen» zu etwas bringen kann. Das genügt nicht. Nur wer die Mathematik genügend schätzt, wer auch terminologisch sauber arbeitet, der hat bei uns eine Chance. Wir ver-

langen von unseren Studenten in den ersten 4 Semestern 40% Mathematik. Mit diesem starken und stabilen mathematischen Hintergrund wollen wir Hochschulinformtiker ausbilden, welche auch in der schnellebigen Informatik Langfristiges erkennen und mitgestalten können. Solche Leute sind in der Praxis dann hoffentlich besonders kompetent, bei Generationsübergängen mitzuarbeiten. Ich sage aber noch einmal, dass nicht sämtliche Leute, die in der Praxis Informatik betreiben, von uns ausgebildet sein müssen.

OP: *Wenn aber die Informatik von der Mathematik her geprägt ist, würde man doch meinen, dass es international gültige Begriffe gibt. Nun aber kann man Abweichungen feststellen. Wie kommt das?*

Prof. Zehnder: Es gibt einmal wesentliche Unterschiede zwischen Mathematik und Naturwissenschaften einerseits und dem Ingenieurgebiet Informatik andererseits. Während es in Mathematik und Naturwissenschaft um die Erkenntnis geht, geht es in der Informatik um Produkte. Das kommerzielle Angebot spielt hier mit, und wir müssen aufpassen, dass dies nicht zu Fehlentwicklungen führt. Zum Beispiel: Vor 10 Jahren bot der Markt erstmals kostengünstige Computer (Home-Computer) an, mit denen man in Basic programmieren konnte. Da damals viele Leute glaubten, dass «Informatik=Programmieren» sei, liessen sich viele Schulen von den Verkäufern verführen; sie glaubten, damit die Informatik in den Schulen eingeführt zu haben. Heute wissen wir, dass das mehrfach falsch war, da das Programmieren in der Schule längst nicht mehr im Vordergrund steht und da sich mit diesen Home-Computern andere wichtige Aspekte der Informatik nicht abdecken lassen. In der Mittelschule darf auch nicht bloss «Computeranwendung» gemacht werden. *Es geht vielmehr um grundsätzliche Verständnisfragen rund um die Informationstechnik und die Darstellung von Informa-*

tion. Warum beschäftigt man sich nicht einmal damit, wie man einen Fragebogen korrekt aufbaut? Solche Fragen müssten in der Mittelschule diskutiert werden. Aber nein, man wurde von der Technik und von der Wirtschaft her praktisch zu einer Schnellentscheidung gezwungen: Jetzt wird programmiert! Da die Informatik eine Ingenieurwissenschaft ist, haben wir diese Probleme. Die Schule hat jedoch die Aufgabe zu überlegen: Was ist *langfristig*? Ich behandle in meinem Unterricht nicht bloss «heutige technische Lösungen» – ausser als Beispiel. Im übrigen – und zwar sage ich das ausdrücklich auf der Stufe Hochschule, wo die Forschung eine viel höhere Kadenz der Erneuerung hat als die Anwendungen in der Praxis – gebe ich meinen Leuten Hinweise auf Zusammenhänge, die in 10 bis 15 Jahren noch Bedeutung haben.

OP: *Und wie finden wir heraus, was in 10 Jahren noch Gültigkeit hat?*

Prof. Zehnder: Wir müssen beachten, dass das technische Angebot, das in den nächsten 10 Jahren auf dem Markt sein wird, heute weitgehend bereits vorhanden ist, in den Forschungslaboratorien und auf dem heutigen Markt. *Wir sind also nicht im Dunkeln, was technisch möglich ist. Wir sind nur im Dunkeln, was sich wirtschaftlich durchsetzen wird.* Und das ist etwas ganz anderes. Die Beispiele Electronic Mail, Fax und Videotex zeigen, dass nicht nur die Technik, sondern sehr wohl andere Faktoren den Markterfolg bestimmen. Wer den Erfolg haben wird, kann die Hochschule nicht sagen. Was ich aber von der Hochschule aus sagen kann ist, dass wir die Augen offen halten müssen für alle Entwicklungen, dass wir uns bemühen müssen, das Richtige zu sehen. Und ich bin überzeugt, dass in den nächsten 10 Jahren die Frage der vernünftigen Gliederung von Grosssystemen – also Föderalisierung – eine ganz grosse Rolle spielen wird. Wir müssen herauskommen aus der Superintegration mit Superabhängigkeiten, wir müssen auch in der Informatik wieder überblickbare Bereiche schaffen, sonst haben wir technische Abstürze. Wir sind heute an der Grenze der zu bewältigenden Komplexität. In der Vorlesung *Gestaltung grosser Informationssysteme* gehe ich Fragestellungen nach wie: Was mache ich, wenn ein System zu gross wird? Wo sind Konflikte? Wo sind Konsistenzprobleme in grossen Datenbanken, und wie kann ich diese beheben? Kann ich sie überhaupt beheben, will ich sie beheben, soll ich gewisse Unausgewogenheiten stehen lassen, weil der Aufwand zu gross ist? Es kann jedoch nicht Aufgabe der Hochschule sein, *alle* zukünftigen Einsatzmöglichkeiten der Informatik dem Studenten zu vermitteln. Es gibt gewisse Dinge, wo die *Erfahrung* im Vordergrund stehen muss und deswegen meine ich, dass die Hochschule solche Überlegungen dann auch in Fortbildungskursen anbieten muss, dass die Leute von der Praxis

Es geht in der Mittelschule vielmehr um grundsätzliche Verständnisfragen rund um die Informationstechnik und die Darstellung von Information.

zurückkommen, um die Problematik etwas systematischer anzuhören. Und ich meine, dass der Hochschullehrer einen Vorsprung hat, weil die Grundsubstanz von kaum jemand sonst aufgearbeitet wird. Und er muss also produktunabhängig seine Konzepte darlegen. Die Didaktikarbeit, die ich leiste für meine Studenten, die ist auch für die Leute in der Praxis von Bedeutung. Im Gegensatz dazu haben natürlich die aktuellen Produkte auf dem Markt auch eine Rolle, aber eine andere. Beides zusammen macht zum Beispiel auch die Bedeutung der Zeitschrift OUTPUT aus. Sie soll den Leuten sichtbar machen, dass wir auch Produkte haben und dass wir uns mit diesen Produkten befassen müssen. *Denn auch Produkte, die man nicht selber einsetzt, können einem etwa helfen, langfristige Entwicklungen zu sehen.*

Aber es braucht auch Leute, welche bewusst nicht produktorientiert denken. Und das ist übrigens für unsere Studenten manchmal schwierig zu verstehen. Wir haben mit Niklaus Wirth einen weltweit bekanntesten Exponenten moderner Programmiersprachen. Die ganze Linie Algol (Rutishauser), Pascal (Wirth), Modula 2 (Wirth) und jetzt Oberon (Wirth), alle diese Entwicklungen hier im Hause, haben unsere Studenten geprägt. Und nun erlebe ich gelegentlich, dass Studenten fragen: «Warum müssen wir so exotische Dinge lernen?» Dann sage ich: «So, jetzt passt bitte einmal auf. Wenn wir Euch Dinge zeigen, die jetzt im Labor von Herrn Wirth entstehen, dann geben wir Euch einen Einblick, wie die Welt in 5 bis 10 Jahren draussen aussehen kann. Und diese Chance, dass wir Euch einen Vorsprung von 5 bis 10 Jahren vermitteln, das ist doch etwas wert. Damit lernt Ihr jetzt schon, Euch Gedanken zu machen über Entwicklungen, die heute hier noch nicht real sind.» Als vor 11 Jahren Herr Wirth seine Lilith gebaut hat, habe ich mit meiner Gruppe diese sofort übernommen. Der Vor-



Also geht es doch darum, dass wir heute für die zukünftige HTL-Informatik vermehrt Maturanden haben sollten. Diese haben ein Loch in ihrer Ausbildung: Es fehlt ihnen die Praxis.

teil war, dass ich diese Entwicklung nicht selber betreuen musste und dass ich 5 Jahre früher als irgendjemand sonst Datenbankentwicklungen graphisch interaktiv auf dem Bildschirm machen konnte. Dieser Vorsprung hat sich ausgewirkt, das können Sie heute sehen, gerade auch in meinen Büchern. Ich muss den Studenten sagen: «Glaubt nicht, irgendjemand an der ETH predige Euch, Oberon sei die künftige Standardsprache der Welt für objektorientierte Programmierung. Aber Ihr müsst daran lernen zu sehen, wo das Wesentliche ist.» Und der Hochschulinformatiker ist in dieser Beziehung bevorzugt gegenüber dem HTL-Absolventen und ganz besonders gegenüber dem Praktiker: Er ist zwar weniger praxisorientiert, aber er sollte imstande sein, sich mit mehreren Generationen zukünftiger Entwicklungen relativ locker zu befassen.

Wir sind also nicht im Dunkeln, was technisch möglich ist. Wir sind nur im Dunkeln, was sich wirtschaftlich durchsetzen wird.

OP: Warum sehen Sie den zukünftigen Informatiker als «Schnittstelleningenieur»?

Prof. Zehnder: Ja, der Hochschulinformatiker muss imstande sein, die ganz grossen Linien zu sehen. Wir müssen heute einen Schritt von den integrierten Grosssystemen in besser gegliederte Teilsysteme machen. Ich habe das in meinem Buch «Informatik-Projektentwicklung» erläutert. Es geht mir darum, dass man lernt, aus integrierten Systemen gegliederte Systeme zu machen, so dass es in Zukunft möglich wird, Teile für sich zu erneuern. Und nur wenn diese Teilerneuerung möglich ist, werden wir in Zukunft überhaupt noch imstande sein, die Erneuerung voranzutreiben. Sonst haben wir am Schluss einen Monolithen, den man nur noch töteln, aber nicht mehr ablösen kann. Für diese Aufgabe brauchen wir relativ wenige Leute, aber mit höchster Übersicht. Diese müssen

nicht nur selber netten kleinen Code für irgendein Spezialproblem schreiben können. Das müssen sie den anderen überlassen, welche nur dies können. *Aber wir brauchen zusätzlich Leute mit einem breiten Horizont, damit sie imstande sind, das Ganze zu gliedern. Und das sind diese «Schnittstelleningenieure».* Das ist eine sehr anspruchsvolle Aufgabe. Er muss klar zuerst lernen, die Detailarbeit zu machen, aber

es wird notwendig sein, vermehrt Leute weiterzubilden mit einer breiten Sicht, also Informatikgeneralisten, welche imstande sind, diesen Generalüberblick zu realisieren und in technische Grenzen umzusetzen, diese Grenzen – Schnittstellen – zu definieren, auszubauen, zu pflegen.

OP: *Zeigt sich hier der Unterschied in der Informatikausbildung zwischen ETH und HTL?*

Prof. Zehnder: Das ist sicher eine Möglichkeit, die dieser Unterscheidung dient. Aber es kommt ein Vorbehalt dazu – und das sage ich ganz offen: Heute haben wir in der Schweiz eine etwas schiefe Aufteilung der Studenten nach den verschiedenen Ausbildungsgängen. Wir haben 1980 begonnen mit der Informatik-Hauptfachausbildung, und gesamtschweizerisch haben wir heute in den Hoch-

schulen Tausende von Informatikstudenten, zusammen mit den Wirtschaftsinformatikern. Daneben haben wir die HTL-Informatikausbildung im Hauptfach, welche wesentlich weniger Leute erreicht. Denn – leider – kennen alle HTL den Numerus Clausus, während an den Hochschulen jedermann mit einer Maturität einfach eintreten kann. Das bedeutet, wir haben also mehr Hochschul- als HTL-Informatiker. Im weiteren sollte man in der HTL eine Berufslehre haben, typischerweise sind das Elektroleute, welche dorthin kommen, aber die ideale Voraussetzung für diese Stufe wäre an sich die Maturität, weil sie wesentlich mehr Mathematik hat als die Berufsschule und weil heute sehr viele Leute in die Maturitätsschulen gehen. *Also geht es doch darum, dass wir heute für die zukünftige HTL-Informatik vermehrt Maturanden haben sollten. Diese haben ein Loch in ihrer Ausbildung: Es fehlt ihnen die Praxis.* Und ich verstehe die HTL-Direktoren, welche sagen: «Wir wollen die Qualität der HTL-Ausbildung nicht aufs Spiel setzen, indem wir diese Praxislinie vernachlässigen.» Aber die HTL Biel kennt eine Lösung, bei der ein Jahr Praxis verlangt wird, aber nicht ein «billiges» Jahr, sondern ein Jahr, in dem man hart arbeitet. Und wenn man abends nicht fertig ist, arbeitet man eben länger. Das ist nämlich auch eine Erfahrung, die die Maturanden nicht ohne weiteres haben. Und diese Entwicklung würde dazu führen, dass man verstärkt den Weg «Maturand HTL-Informatik» anbieten sollte.

OP: *Was ist der Informatik-Föderalismus? Wie funktioniert dieser?*

Prof. Zehnder: Der funktioniert analog zum politischen Föderalismus. Er basiert darauf, dass jeder Teilbereich alles selber macht, was er selber erledigen kann und dass er erst zum nächst oberen geht, wenn er es nicht mehr allein machen kann. Das ist die Grundidee. Und das heisst im Informatikbereich wiederum, Teile abzugrenzen, welche weitgehend

EIGNER+PARTNER

DATA VIEW

DAS SOFTWAREWERKZEUG ZUR ERSTELLUNG VON DATENBANK-APPLIKATIONEN

- datenbankunabhängig
- hardwareunabhängig
- alphanumerisch und graphisch
- objektorientiert

DESKTOP-OBERFLÄCHE
 GRAPHISCHE WINDOWS (NOTIF/ X.11)
 NEUTRALE WINDOW-SCHNITTSTELLE
 PROGRAMM-SCHNITTSTELLE
 DATENSCHUTZ/ZUGRIFFSKONTROLLE
 REPORT-GENERATOR
 DATENBANK-NEUTRALE SCHNITTSTELLE EP-SQL
 MASKEN-/MENU-GENERATOR
 OBJEKT-GENERATOR
 RDBMS ROB, ORACLE, INGRES
 SCO
 VMS
 UNIX
 HARDWARE IBM, DEC, HP, SUN

EIGNER + PARTNER GmbH, Ruschgraben 133, W-7500 Karlsruhe, Tel. (0721) 62 91-0, Fax (0721) 62 91 88
 EIGNER + PARTNER AG, Bahnhofstr. 693, CH-4622 Egerkingen, Tel. (062) 61 20 91, Fax (062) 61 26 12
 Weitere Geschäftsstellen in Dresden und München

autonom sind und dass ich nur dann zum Rest Anschluss suche, wenn es notwendig ist, und nicht umgekehrt. Also ich beginne nicht von oben und gliedere nach unten, das klassische Top down, sondern ich gehe davon aus, dass in der Praxis viele Bottom-up-Situationen vorhanden sind. Als Datenbankspezialist und -theoretiker müsste ich oft sagen: «Dort, wo Daten gemeinsam genutzt werden können, zusammenlegen!» Das ist gut und recht. Aber jetzt kommen wir in Bereiche, die so gross geworden sind, dass das Zusammenlegen neue Gefahren aufweist. Deswegen meine Antwort: *Föderalismus ist auch hier eine sinnvolle Grundkonzeption.*

OP: *Bei der Entwicklung von Projekten braucht es spezielle Methoden, und alle Welt spricht im Moment von Objektorientiertheit. Braucht es ein Redesign der Datenbankwelt, wenn diese Objektorientiertheit hineinkommt oder kann man das einfach so implementieren, anhängen?*

Prof. Zehnder: Also, man muss sich einmal im klaren sein, dass diese modernen Techniken alle auch ihre Entwicklungen haben. Ich habe 1966 begonnen, in einer Frühform der Simulationsprache Simula zu arbeiten. Bereits in Simula 67 existierten das Klassenkonzept und andere Dinge, die heute in objektorientierten Sprachen drin stehen. Es ist nicht alles neu, was hier plötzlich so aussieht. Und ich bin voll überzeugt, dass diese Datenkapselung, die beispielsweise in der objektorientierten Programmierung eine Rolle spielt, ein kräftiges Werkzeug ist, um bessere Programme zu schreiben. Das ist recht und gut. Aber, das heisst überhaupt nicht, dass die grossen Informationssysteme primär auf dieser Ebene konzipiert oder gelöst werden, sondern da geht es einmal darum, diese Informationssysteme als solche zu gestalten und zu gliedern: Welche Informationen sind in einer Firma von Bedeutung heute, in 5 Jahren, in 10 Jahren? Und wie werden gewisse Informationen ausgetauscht?

Das ganze Archivproblem ist ja noch längst nicht gelöst. Und erst jetzt kommt die Frage: Wie wird das physisch gelöst, physisch-logisch, hardware- und softwaremässig? Aber das sind schon Fragen zweiter Ordnung. Und selbstverständlich ist uns alles recht, was dazu führt, bessere Software zu schreiben. Software ist nicht einfach etwas, was auf einem Computer läuft, sondern was richtig läuft. **OP:** *Gibt es beim Entity Relationship Model eigentlich das Vererbungsprinzip?*

Prof. Zehnder: Schauen Sie, das ist wieder eine Frage der Terminologie. Wenn man das Modell so erweitert, dass man umfassende Datenmengen aufnehmen kann, ist die «Vererbung» möglich. Bei überlappenden Entitätsmengen muss man Überbegriffe einführen. Und genau da gibt es dieses Vererbungsprinzip, nur läuft es hier nicht mit dem Namen Vererbung. Das Wort Vererbung kommt in der herkömmlichen Datenbankwelt noch nicht vor. Aber diese Konzepte sind keineswegs so neu. Und wenn man das spürt, empfinde ich es zum Beispiel als eine Aufgabe meinerseits, auf diese Situation hinzuweisen. Auch gute technische Konzepte brauchen oft sehr lange, bis sie sich durchsetzen können; ich habe bereits das Klassenkonzept erwähnt. Es gibt an vielen Orten gute Ideen. Ich halte es für einen Vorzug des Hochschulmenschen, dass er eben herumschauen darf, er, der nicht geprägt ist durch eine einzige Firmentradition. Wir müssen diese Vorteile übernehmen. Darum richte ich meinen Unterricht auf 10 bis 15 Jahre aus. Es sollte aus meinen Büchern ganz deutlich hervorgehen, dass ich so denke. Unix ist beispielsweise ja auch schon 21 Jahre alt.

OP: *Im Namen der Output Redaktion möchte ich herzlich danken für dieses Gespräch und wir wünschen Ihnen selbstverständlich weiterhin noch viel Erfolg bei der Realisierung Ihrer Anliegen.*

”ACHTUNG“

Anwender von Canon Tischlaserdruckern

(oder kompatiblen Geräten)

Statt 3 original
Canon-Kartuschen
für 9000 Drucke
brauchen Sie jetzt
nur noch eine berolina
SUPERCART.



- Sie sparen
Kosten
- verbessern
Ihr Schriftbild
- entlasten
die Umwelt

Sofort-Info:

055/63 27 67

oder
datadesign AG, berolina Schriftbild, Letzi-
strasse, 8852 Altendorf, Fax: 055/63 57 03

berolina
...das
Schriftbild