Zeitschrift: Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

Herausgeber: Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

Band: 71 (1980)

Heft: 11

Artikel: Software-Ausbildung : Wettlauf mit der Zeit

Autor: Zehnder, C. A.

DOI: https://doi.org/10.5169/seals-905259

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Mehr erfahren

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. En savoir plus

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. Find out more

Download PDF: 16.07.2025

ETH-Bibliothek Zürich, E-Periodica, https://www.e-periodica.ch

Software-Ausbildung: Wettlauf mit der Zeit

Von C. A. Zehnder

681.3.06:331.86(494);

Die technische Entwicklung in der Digital-Hardware verlief derart rasch, dass ihr die Umgebung (Personal, Ausbildung, Problemverständnis usw.) nicht folgen konnte. Nun müssen die Lücken kurz- und langfristig geschlossen werden. Der Artikel zeigt für schweizerische Verhältnisse, dass auch für eine vor 30 Jahren völlig unbekannte Tätigkeit wie die Software-Entwicklung die vorhandene Schulungsinfrastruktur zweckmässig miteinbezogen werden muss.

Le développement du matériel en informatique a été si rapide, que la formation du personnel et sa compréhension des problèmes pratiques accusent un très net retard. Il s'agit donc d'y remédier au plus vite. L'auteur montre que, désormais, l'enseignement en Suisse doit tenir compte convenablement du logiciel, discipline qui était totalement inconnue il y a une trentaine d'années.

1. Stellenwert der Software

Eine der faszinierendsten Eigenschaften der Digitaltechnik ist ihre beinahe universelle Verwendbarkeit: Die gleiche Hardware kommt für verschiedenste Anwendungsprobleme in verschiedenster Art und Weise zum Einsatz. Natürlich gibt es Ausnahmen, wie etwa beim spezialisierten Uhrenmikroprozessor. Aber in den meisten Anwendungsfällen schafft erst die Software die Querverbindung zwischen Hardware und Anwendung, beim Mikroprozessor wie beim Grosscomputer.

Anwendungen können nur korrekt von einem Rechner unterstützt werden, wenn ihre Probleme genau erkannt sind. Das gilt für technische Applikationen so gut wie für administrative. Es ist aber offensichtlich, dass jene Leute rar sind, welche über profunde Anwendungskenntnisse verfügen und erst noch in der rasenden Entwicklung der Digitaltechnik zuhause sind. Ohne solche Leute ist jedoch der Hardwareeinsatz nicht möglich oder er führt zu unvernünftigen Fehlentwicklungen. Dabei liegen noch ungeahnte sinnvolle Anwendungen offen in der Zukunft, sie müssen nur erkannt werden.

2. Software-Konstruktion

Die Entwicklung von Programmen war bei den ersten Computern eine Art von Kunst, denn der gute Programmierer schuf sein optimales Programm mit einer Mischung von Intuition, scharfem Nachdenken und Fleiss. Beschreiben liess sich dieses Vorgehen nicht rational, und es liess sich vor allem nicht schulmässig lernen. Aber man brauchte optimale Maschinenprogramme, sonst konnte man den Computer gar nicht einsetzen, denn die technischen Leistungen der frühen Rechengeräte waren sehr begrenzt. Typische Arbeitszeiten für die ERMETH, Rechenmaschine der ETH 1955...63, waren beispielsweise Programmläufe von 1...2 h für eine einfache Rechnung, die heute in 1 s erledigt ist.

Die leistungsfähigere Transistor-Hardware der sechziger Jahre ermöglichte nicht nur mehr und kompliziertere Applikationen, sie verhalf auch den höheren Programmiersprachen zum Durchbruch. Langsam bürgerten sich FORTRAN, ALGOL, COBOL, PL/I ein, und die Assembler-Programmierung erhielt ernstzunehmende Konkurrenz; eliminiert war (und ist) sie aber noch nicht. Immerhin hatte man erkannt, dass die Verwendung höherer Programmiersprachen und der entsprechenden Compiler die Voraussetzung für systematischere Software-Herstellung war, wobei gleichzeitig Programmierfehler vermindert und Programme portabel (übertragbar von einer Maschine auf eine ähnliche) gemacht werden konnten. Für diese Vorteile waren auch eifrige Assembler-Programmierer nach viel Überredungskünsten langsam bereit, auf höhere und damit *etwas* weniger laufzeiteffiziente Programmiersprachen umzusteigen.

Die Entwicklung geht aber noch weiter. Moderne Applikationssysteme führen immer näher an die Anwendung heran. Ihre Benützungssprache ist die Sprache des Bauingenieurs oder des Wertschriftenanalytikers, ganz nach Bedarf. Programmgeneratoren und andere, auch methodische Hilfsmittel verkürzen die Programmentwicklung stark. Als Tendenz lässt sich festhalten, dass die Software immer mehr den Charakter eines typischen technischen Produkts erhält, dessen Herstellung man ebenfalls technisch unterstützen kann. Die «genialen Einfälle» des Programmierers der Frühzeit sind nicht nur obsolet, sondern schädlich geworden, da sie nämlich die systematische Wartung und eine eventuelle nachträgliche Anpassung von Programmen praktisch verunmöglichen.

Diese Aussagen gelten heute für Grosscomputer ziemlich unbestritten. Vor wenigen Jahren kamen nun die Minicomputer auf. Die Programmierer der Minis arbeiteten im Assembler-Code. Höhere Programmiersprachen für Minis? Undenkbar, aus Effizienz- und vielen anderen Gründen! In der Zwischenzeit haben die höheren Sprachen mit den entsprechenden Compilern auch den Minibereich erreicht. Sie sind gut und reif, wie die Beispiele PASCAL und MODULA aus der Arbeitsgruppe von *N. Wirth* (ETHZ) zeigen. Zusätzlich sind Entwicklungshilfen entstanden, welche die Bequemlichkeiten eines Grosscomputers für die Entwicklungsarbeit verfügbar machen, worauf das fertig entwickelte Programm automatisch in die Maschinensprache des Mini übersetzt und dort zur Ausführung gebracht wird (Cross-Compiler).

Nach diesem Exkurs in die Welt der Minicomputer liegt die Fortsetzung für die Mikros auf der Hand: Auch bei den Mikroprozessoren gehört die Zukunft den höheren Programmiersprachen, den Entwicklungshilfen und den Programmgeneratoren. Das bedeutet, dass die Grundausbildung für einen Software-Ingenieur der verschiedensten Digitalrechner durchaus grosse Ähnlichkeit aufweisen darf und soll. Auf all diesen Geräten geht es nämlich darum, Prozesse zu formulieren, zu strukturieren und in gegenseitige Abhängigkeit zu bringen; und es geht um die Strukturierung von Datensystemen. Selbstverständlich muss man sich dabei auch um die technischen Gegebenheiten der zur Verfügung stehenden Hardware kümmern. Aber ändert gegenwärtig etwas schneller als diese Hardware? Also konzentriere man sich auf das Grundlegende. Software-Konstruktion als saubere Ingenieurtätigkeit sei das Ausbildungsziel.

Vortrag der SEV-Informationstagung «Der Mikroprozessor als Problemlöser und Problemquelle», 29. April 1980 in Neuchâtel.

3. Grundausbildung und Spezialkenntnisse

Das Gesagte führt zur klaren Trennung von Grund- und Spezialausbildung.

Die *Grundausbildung* (Berufsausbildung, Mittelschule, aber auch Hochschule und Ingenieurschule) muss grundsätzliche Einsichten in Programmabläufe, Prozesse und Datenstrukturen sowie die Methoden zur Software-Entwicklung vermitteln. Spezielle Geräte und Maschinensprachen werden nur behandelt, wenn dies für praktische Arbeiten und als Beispiel nötig ist. Es kann nicht die Aufgabe der Grundausbildung sein, für ganz bestimmte Geräte, welche zufällig heute auf dem Markt sind, Detailkenntnisse zu vermitteln. Auf der anderen Seite ist auch klar, dass man nicht bloss theoretisieren soll und daher Anwendungen auf realen Geräten beispielhaft durchführen muss.

Spezialkenntnisse müssen dauernd im Laufe der Berufstätigkeit erneuert werden können. Das gilt für Kenntnisse im Bereich der Informatik, aber auch in der Anwendung. Wer auf dem Gebiete der Softwareentwicklung arbeitet, kommt nicht darum herum, sich laufend um Neuerungen zu kümmern. Das muss aber nicht heissen, dass jeder jedes Marktangebot kennen muss. Daher wird z. B. die Beschaffung der Hardware auf spezialisierte Stellen konzentriert.

4. Informatik als Hauptberuf oder als Ergänzung

Sehr viele Leute brauchen in ihrer Berufstätigkeit Mathematik, nur wenige sind Mathematiker. Analog brauchen bereits viele Menschen Kenntnisse in Informatik (und ihre Zahl nimmt noch laufend zu), doch müssen sie nicht alle hauptberufliche Informatiker sein. Gerade die Interdisziplinarität ist wichtig, damit Digitaltechnik und Anwendung kombiniert werden können.

Das Beispiel der Mathematik sei aber noch etwas weiter betrachtet: Ohne einen Stamm von Mathematikern kann die Mathematik nicht genügend vermittelt werden. Dabei gehören zum Stamm nicht nur Lehrer, sondern auch Forscher, angewandte Mathematiker, Operations-Research-Leute usw. Ähnlich sollte es auch in der Informatik sein. Dazu braucht es eine Anzahl von *Fach-Informatikern,* welche in fachlicher und methodischer Hinsicht sicher sind und in ihrer Hauptdisziplin ständig am Ball bleiben. Diese Fachinformatiker arbeiten in der Hochschule und in der Systemprogrammierung; sie gehören auch in die eigentlichen Projektteams hinein, welche grössere Applikationssysteme entwickeln.

In diesen Teams und an vielen anderen Orten braucht es auch Fachleute der Anwendung, Ökonomen, Ingenieure usw. mit einer soliden *Nebenfachausbildung* in Informatik. Eine solche Nebenfachausbildung müsste mindestens gute Kenntnisse in Programmiermethoden und im Grundaufbau von Computersystemen umfassen, ergänzt durch einzelne Vertiefungsfächer, wie z. B. Kommunikationstechnik, Datenbanksysteme, oder ähnliche. Diese Kenntnisse sollten mit ersten Eigenerfahrungen ergänzt werden, wie sie in Semesterarbeiten und Praktika erworben werden können. Nebenfach-Informatik ist eine Ergänzung, welche die Grund-Ingenieurausbildung irgendwelcher Richtung heute in vielen Fällen erst richtig zum Tragen bringt. Die Nebenfach-Informatik-Ausbildung kann während des Grundstudiums oder auch erst später in einem Aufbaustudium erworben werden. Diese letztere Lösung dürfte gerade für die nächsten paar Jahre von besonderer Bedeutung werden.

Haupt- und Nebenfachausbildung können nicht vollständig unabhängig voneinander angeboten werden. Eine gute Nebenfachausbildung, wie sie hier postuliert wird, ist nur realisierbar, wenn genügend Hauptfachinformatiker zur Verfügung stehen. Sie bilden den Kristallisationskern mit der Substanz für die weitere Ausstrahlung.

5. Ausbildungssituation in der Schweiz

Im schweizerischen Berufsbildungssystem nimmt die Informatik heute einen Platz ein, der in keiner Weise ihrer wirtschaftlichen Bedeutung entspricht. Fachgebiete mit Tradition (etwa das Bauwesen oder die Elektrotechnik) verfügen über ein volles berufsbildendes Sortiment von Ausbildungsgängen auf Stufe Berufslehre, HTL und Hochschule, zusätzlich über höhere Fachprüfungen (Meister) und Nachdiplomkurse verschiedener Art. In der Informatik ist dieses System noch völlig anders, zum Teil im Aufbau, zum Teil nicht existent.

Hochschulen: Von den zehn schweizerischen Hochschulen haben alle ausser Basel einen oder mehrere Lehrstühle für Informatik (ETHZ: 6). Damit kann überall Informatik «im Nebenfach» studiert werden, wobei je nach Hochschule das Hauptfach anders liegt.

In neuester Zeit traten auch *Hauptfach*-Informatik-Lehrpläne in Kraft (Fribourg, Genf), oder sind in Vorbereitung (Bern, Zürich, EPFL, ETHZ).

Im Rahmen der Hochschulen muss erwähnt werden, dass heute ein Grossteil der Ingenieure, Naturwissenschafter und Ökonomen (also jährlich über 3000 Studenten) eine Grundausbildung im Computereinsatz erhält, was für ihre Berufstätigkeit als selbstverständlich betrachtet wird.

Höhere Lehranstalten: Die Situation an den HTL ist ähnlich wie an den Hochschulen, indem der Computer seit Jahren vielerorts einen Lehrgegenstand und ein Lehrmittel darstellt. Dennoch sind eigentliche Informatik-Lehrgänge grosse Ausnahmen:

– Am Neu-Technikum Buchs wurden nebenbei von 1975–1979 total 15 Maturanden(!) zu Informatikern ausgebildet (3-Jahres-Kurse).

– An den Ingenieurschulen Yverdon (ab 1979/80) und Biel (ab 1980/81) sind Informatiker-Klassen definitiv bewilligt und im Anlaufen. Interessanterweise werden dabei nicht nur technische, sondern ausdrücklich auch Wirtschaftsinformatiker ausgebildet.

Die Höheren Wirtschafts- und Verwaltungsschulen (HWV) vermitteln zwar Kenntnisse aus der Informatik, aber nicht im Hauptfach.

Berufslehre: Eigentliche Berufslehren für «Informatiker» existieren nicht. Innerhalb der kaufmännischen Berufsausbildung haben sich jedoch gewisse Vertiefungsrichtungen herausgebildet. Dabei werden im Rahmen der Vertiefung «Rechnungswesen» auch Kenntnisse zur Büroautomation vermittelt.

Hauptfach	Nebenfach	Hochschulen
Mathematik oder Naturwissenschaften	Informatik	Bern, Genf, Lausanne, Neuenburg, Zürich, EPFL, ETHZ
Ökonomie		Freiburg, Genf, Lausanne, Neuenburg, St. Gallen, Zürich

Dies wären die regulären Berufsbildungsgänge. Die heute im Informatikbereich Tätigen haben ihre Ausbildung jedoch fast ausschliesslich anders beschaffen müssen. Auch dazu einige Hinweise:

Kurse für Praktiker: Seit Beginn der kommerziellen elektronischen Datenverarbeitung bilden die Kurse der Computerhersteller einen zentralen Ausbildungsschwerpunkt in Informatik (wobei dieser Begriff damals noch nicht existierte). Heute sind zu den Computerherstellern weitere Kursanbieter getreten, von Privatschulen über Softwarehäuser bis zu Fachverbänden. Dabei ist die Ausbildung heute in allen Fällen eine fakturierte Dienstleistung und nicht mehr, wie bis in die sechziger Jahre, eine Art Reklamegeschenk. Die Zahl der externen Kursbesucher ist seit mehreren Jahren eher stationär, aber weiterhin von grosser Bedeutung. Als Hinweis auf die Grössenordnung diene die Zahl von 20000 jährlichen Studententagen im Rahmen der Kundenschulung von IBM Schweiz.

Software-Schule Schweiz: Wegen der bisher fehlenden Fachinformatiker-Ausbildung auf Stufe ETH/HTL wurde im Sinne einer Übergangslösung für die Jahre 1979–1983 mit Bundesgeldern die Software-Schule Schweiz geschaffen. Der Industrie sollen kurzfristig Software-Ingenieure verschafft werden, indem jährlich zwei 7-Monat-Kurse mit je 30 gut vorgebildeten Teilnehmern durchgeführt werden.

An einer einzigen Stelle im Bereich der Berufsausbildung weist auch die Informatik seit einigen Jahren schweizerische Normalverhältnisse auf, und zwar bei den sog. *Höheren Fachprüfungen* für EDV-Analytiker (Eidg. Diplom, BIGA). Allerdings werden damit kaum Fachleute im technischen Anwendungsbereich angesprochen, sondern fast ausschliesslich solche aus kommerziell-administrativen Applikationen. Die berufsbegleitende Vorbereitung für diese Prüfung dauert mehrere Jahre und wird von verschiedenen Verbands- und Privatschulen angeboten. Die Prüfung selbst zerfällt in Vor- und Schlussprüfung. Die Absolventenzahlen der Schlussprüfung betragen nach BIGA-Angaben:

Schlussprüfung	1976	1977	1978	1979	total
Kandidaten	19	52	83	144	
Diplomierte	6	24	51	63	144

Von den bisher 144 Diplomierten sind übrigens bloss 4 Frauen!

Aus leichtverständlichen Gründen kann die für das Informatik-Gebiet wohl wichtigste Ausbildungsform, nämlich das *Training am Arbeitsplatz,* nicht quantitativ charakterisiert werden.

6. Die zeitliche Diskrepanz

Die Berufsbildungs-Infrastruktur ist in der Schweiz im Bereich der Informatik ungenügend; sie ist aber im Aufbau. Schüler und Studenten sind bei der Berufswahl sehr konservativ und besuchen kaum echt «freiwillige» Fächer (weshalb z.B. die höheren Software-Kurse an der ETH Zürich seit 10 Jahren nur ganz wenige Teilnehmer aufweisen). Sobald aber ein «offizielles» Informatik-Studium, mit Abschluss und Titel existieren wird, werden auch die Studenten kommen. Damit werden in einigen Jahren junge Berufsanfänger mit Informatik-Grundausbildung vorhanden sein.

Aber was soll in der Zwischenzeit geschehen? Die erwähnte Software-Schule Schweiz wurde geschaffen, weil die Industrie zu wenig Software-Ingenieure hat und *jetzt* damit in Schwierigkeiten gerät, weil die Mikroprozessoren *jetzt* Eingang im Apparatebau halten. Wenn die Industrie einen wettbewerbskritischen Engpass verspürt, kann sie sicher nicht vier Jahre warten, bis ein paar (sicher aber zu wenig) Maturanden die richtige Nase für das vielversprechende Studium gehabt und das ganze Studium hinter sich gebracht haben.

Die Frage der kurzfristigen Anpassung und Ausbildungsergänzung ist somit nicht allein dadurch bedingt, dass die Informatik am Anfang steht. Auch in Zukunft muss die Industrie momentane Engpässe durch Kurse abdecken. Je besser die Grundausbildung der verfügbaren Leute ist, desto leichter ist dies möglich. Die Anpassung der Grundausbildungsgänge an Spezialbedürfnisse kommt immer zu spät. Hingegen ist der Beizug von Hochschulen und HTL für Kurse sehr zweckmässig.

7. Schlussfolgerungen

Die meisten in der Informatik Tätigen in der Schweiz haben ihre Fachkenntnisse am Arbeitsplatz erworben. Das war bisher unumgänglich, und es wird weiterhin eine zentrale Ausbildungsmöglichkeit bleiben. Viele grössere Firmen haben denn auch ihre interne Schulung gut ausgebaut und eigene Lehrkräfte eingesetzt.

Das darf aber nicht dazu führen, dass die Informatik-Gemeinschaft bezüglich Grundausbildung resigniert. Die Informatik ist im schweizerischen Ausbildungssystem bisher ungenügend integriert. Es werden Grund-Ausbildungsgänge auf allen Stufen benötigt (Hochschule, HTL/HWV, Berufslehre) und in verschiedenen Anwendungsgebieten. Dabei sei aber in einer Beziehung gewarnt: Die Grundausbildungen dürfen nicht kleinlich auf Anwendungsbereiche zersplittert werden. Ein Wechsel des Anwendungsgebietes muss für einen Informatiker immer relativ offen bleiben. Auch das ist eine Möglichkeit der so zentral wichtigen Weiterbildung.

Informatiker brauchen eindeutig mehr als blosse Fachkenntnisse auf ihrem Spezialgebiet, denn Informatik ist vor allem ein Instrument für Anwendungen. Deshalb benötigt der Informatiker Verständnis für die Applikation und Fähigkeit zur Zusammenarbeit. Die Ausbildung dafür ist mindestens so schwierig wie diejenige der Fach-Informatik selber, jedoch ebenfalls ein erreichbares Ziel.

Adresse des Autors
Prof. Dr. *C.A. Zehnder,* Institut für Informatik, ETH-Zentrum, 8092 Zürich.