

## **Informatiker müssen mehr als Programme schreiben.**

### **Die Softwareentwicklung ist neu zu strukturieren.**

In der letzten Zeit häufen sich die Informationen darüber, daß wegen mangelhafter Software z.B. moderne S-Bahnen nicht zum Einsatz kommen, Visa nicht rechtzeitig zur Urlaubsreise erstellt werden können, ein Systemwechsel das Sozialamt zum Pflegefall macht etc.

Die wichtigsten Ursachen für derartige Pannen sind, daß die Informatiker die notwendige Funktionalität der Software nicht, nicht richtig oder nicht vollständig realisiert haben.

Analysiert man das Berufsbild des Informatikers, so stellt man fest, daß wichtige Tätigkeiten z.Zt. in dem Beruf vereint sind:

Die Funktionalität, die in einer Software abgebildet ist, muß beschrieben werden und die Beschreibung ist in eine Sprache zu übersetzen, die vom Computer verstanden und ausgeführt werden kann (Programm).

Die Ausbildung der Informatiker ist heute schwerpunktmäßig auf die Methodik und Technik der Softwareentwicklung ausgerichtet, die Übersetzungstechnik wird gelehrt. Dabei wird vor allem eine sequentielle Denkweise gefordert.

Die Ermittlung und Beschreibung der Funktionalität einer Software verlangt im Gegensatz dazu Kenntnisse in dem Anwendungsgebiet und ein vernetztes Denken, um die Zusammenhänge der komplexen Realität zu erkennen und zu berücksichtigen.

In den Anfangsjahren der Datenverarbeitung eine Trennung von Organisation und Programmierung normal. Über den Organisationsprogrammierer entwickelte sich dann das Berufsbild des Informatikers, bei dem die organisatorische Kompetenz wenig gefördert wird. Die Komplexität der modernen Softwareprodukte sowohl im Hinblick auf den fachlichen Umfang als auch auf die technische Vernetzung fordern eine Trennung der fachlichen Verantwortung wie im Hausbau: Bauherr, Architekt, Statiker und Handwerker.

**Der Bauherr:** Dem Bauherrn entspricht im Softwarebereich der Anwender, der eine Softwareprodukt benötigt. Wenn er nicht in der Lage zu beschreiben, welche Funktionalität er benötigt, so kann er nicht erwarten, daß er Entwickler oder Lieferant eine Software liefert, die genau seinen geheimen Wünschen und Vorstellungen entspricht. Auch der Architekt muß wissen, was der Bauherr will, damit er verbindliche Pläne vorlegen kann.

**Der Architekt:** Er ist mit dem Organisator zu vergleichen, der die Bedürfnisse des Anwenders analysieren, strukturieren und beschreiben kann. So wie der Architekt sowohl die baurechtlichen Anforderungen als auch die letzte vom Bauherr geforderte Steckdose im Bauplan auszuführen hat, muß der Organisator verbindlich beschreiben, welche Funktionalität in der Software realisiert werden muß.

Der Statiker: Dem Statiker entspricht dem Systemplaner, der auf der Grundlage der Funktionsbeschreibung ermittelt, auf welchen Hardwarekomponenten und mit welchen Methoden und Techniken die Software zu realisieren ist.

Der Handwerker: Wie der Handwerker im Hausbau hat der Programmierer auf der Grundlage der Funktionsbeschreibungen und den Anweisungen der Systemplaner die Programme zu erstellen und funktionsfähig bereitzustellen.

Gegen diese Funktions- und Verantwortungstrennung wird sicherlich eingewendet werden, daß die Softwareentwicklung nicht mit dem Hausbau zu vergleichen sei. Vor allem sei nie die Zeit vorhanden, um schrittweise vorzugehen. Man müsse so schnell wie möglich mit der Programmierung beginnen, um die knappen Termine erfüllen zu können. Diese Verhaltensweise, von der die meisten Softwareverantwortlichen wissen, daß sie falsch ist, führt dann zu den Ergebnissen, die zu Anfang beschrieben ist. Sicherlich ist das Produkt Software nicht mit dem Produkt Gebäude zu vergleichen. Aber für alle Produktionen gelten unabhängig von ihrer Art die Arbeitsschritte: Produktanforderung erstellen, Machbarkeit prüfen und Arbeitsvorgaben erstellen und Produktfertigung.

Solange die Informatiker sich mit kreativen Künstler vergleichen, die in ihrem Produkt sich selbstverwirklichen, ist es nicht verwunderlich, wenn der Anwender seine Anforderungen in dem Produkt nicht wiederfindet.

Wenn Sie die Verantwortlichen in der Softwareentwicklung auf derartige Mißstände hinweisen, so findet man in der Regel Zustimmung. Sie wissen alle Bescheid über die Mängel, die zu den wenige erfreulichen Ergebnisse führen. Sie kennen die Bedeutung von Pflichtenheften, Dokumentationen, Planungsqualität, planvollen Tests etc. Aber alle diese Erkenntnisse werden dem Moloch Termin geopfert. Enge Termine selbst müssen noch nicht zu schlechten Produkte führen. Nur muß in solchen Fällen der Funktionsumfang der Aufgabenstellung rechtzeitig den Terminen angepaßt werden. Die Einstellung: "Mal sehen wie weit wir kommen" wird mit Sicherheit zu schlechten Ergebnisse und unzufriedenen Kunden führen. Auch den Unternehmensleitungen muß verdeutlicht werden, daß in der Softwareentwicklung fast alles machbar ist, jedoch nicht in jedem beliebigen Zeitraum.

-----  
Die Funktionalität ist in der Regel gleichzusetzen mit realen Vorgängen, Zuständen und Regeln. So ist bei z.B. bei einem Kaufvorgang folgendes zu beobachten: Tausch von Ware gegen Geld, Veränderung von Waren- und Kassenbestand und der Preis gibt die Regel für den Tausch an.

Das Berufsbild des Informatikers fordert drei wichtige Eigenschaften:

1. er muß die methodischen und softwaretechnischen Kenntnisse der Softwareentwicklung beherrschen
2. er muß in der Lage sein, seine Arbeit optimal zu organisieren und

3. er muß die realen Inhalte seiner Programme genau kennen und beschreiben können.

Beobachtet man die Mitarbeiter in den Softwareabteilungen, so kann man folgendes feststellen: Punkt 1. wird auf Grund der Ausbildung durchweg beherrscht, im Punkt 2. unterscheiden sie sich nicht wesentlich von anderen Bereichen in den Unternehmen und Punkt 3. erlernt er erst im Laufe einer langjährigen Erfahrung, wenn überhaupt.