

Von den Software-Entwi

KOMMENTAR



Dr. Roland Drewinski, Mitglied der Geschäftsleitung bei der Contact Software GmbH in Bremen

Software-Entwicklung und klassische Ingenieurdisziplinen wie Maschinenbau oder Elektrotechnik haben mindestens eines gemeinsam: zielsicher Produkte zu entwickeln. Ansonsten scheint es sich aber um getrennte Welten zu handeln. Bislang haben die Softwareentwickler eher neidvoll auf die etablierten Ingenieure und ihren Fundus an Wissen und Methoden geblickt: Software sollte nicht länger auf Kunstfertigkeit beruhen, sondern gleichermaßen Ergebnis methodischen Vorgehens sein.

Die Software-Technik ist eine junge Disziplin. Gebeutel durch immense Schwierigkeiten bei der Entwicklung großer Softwaresysteme wurde der Begriff »Software Engineering« 1968 auf einer NATO-Konferenz geprägt. Der Weg sollte weg vom Individualisten und hin zur kooperativen Entwicklung im Team führen. Angelehnt an das Vorbild der klassischen Produktion entstand schließlich auch die Idee der »Software Factory«.

Getrieben durch immer anspruchsvollere Aufgabenstellungen und die andauernden Schwierigkeiten in vielen Projekten wurde in den folgenden Jahren und Jahrzehnten mit immensem Aufwand weltweit nach besseren Programmiersprachen, Entwicklungsumgebungen, Vorgehensmodellen oder Qualitätsrichtlinien gesucht. Resultate sind zum Beispiel das V-Modell für das systematische Projektvorgehen oder das »Capability Maturity Model« zur Beurteilung des Software-Entwicklungsprozesses.

Fragen zur Systematik des Entwicklungsprozesses stehen in Bereichen wie dem Maschinenbau, in denen »richtige« Produkte entwickelt werden, weniger hoch im Kurs. Im Mittelpunkt sehen viele Unternehmen unverändert die kreativen Fähigkeiten des einzelnen Ingenieurs,



der sich am Fundament der Physik mit seinen klaren Randbedingungen orientiert. Produktanforderungen lassen sich meist durch vergleichsweise wenige, klar benennbare Leistungsparameter beschreiben.

Modellbildung und Arbeitsteilung erfolgen in der Regel orientiert an Komponenten und ihren Zusammenhängen. Mit der visuellen Beschreibung in 2D und 3D, einer ausgereiften Zeichnungssymbolik,

cklern siegen lernen?

zahlreichen etablierten Standards und einer umfassenden Terminologie steht ein anerkanntes Repertoire zur Dokumentation der Ingenieursarbeit zur Verfügung.

Software ist dagegen tatsächlich »weiche Ware«: Anforderungen lassen sich auf vielen Wegen an der Benutzerschnittstelle abbilden. Befreit von physikalischen Gesetzmäßigkeiten steht ein ungleich größerer Lösungsraum zur Verfügung, der in aufwändigen Abstimmungsprozessen mit Mitteln wie beispielsweise »Requirements Management« oder »Rapid Prototyping« eingegrenzt werden muss.

Schaut man genauer hin, wird deutlich: Die Zeiten ändern sich. Heute haben die klassischen Ingenieure mindestens ein paar gute Gründe, umgekehrt auch neidvoll auf die Software-Entwickler zu blicken:

Die Mechatronik kündigt vom Zusammenwachsen der Disziplinen. Produktentwicklung heißt zusehends verteiltes Arbeiten und erfordert, sich besser zu organisieren und Teamwork an die erste Stelle zusetzen. Die Reduzierung der Durch-

laufzeiten, »Mass Customization« und Reglementierungsflut (Stichwort »Compliance«) erfordern systematisches Projektvorgehen, um Zeit, Kosten und Qualität im Griff zu behalten.

Von den Ingenieuren wird erwartet, dass sie ihre Kreativität nun mit der Fähigkeit zur systematischen Produktentwicklung im Team kombinieren! Fragen nach der Effizienz und Effektivität der Projektarbeit

100 Prozent in proprietären CAX-Dateiformaten vor. In der Software-Industrie dagegen stehen die Arbeitsergebnisse im Quelltext zur Verfügung.

Der Marktanteil »proprietärer« Programmiersprachen beträgt circa 20 Prozent, der Marktanteil »unzugänglicher« binärer Datenformate quasi 0 Prozent. Andere Pluspunkte auf Seiten der Software-Technik sind dort fest etablierte Themen wie Konfigurations-, Anforderungs- und das so genannte Release-Management.

Mit PDM/PLM-Systemen stehen Lösungen zur Verfügung, die den Anspruch haben, das Beste aus beiden Welten zu verbinden. Durch Trends wie Mechatronik wird der Druck noch zunehmen, diesen Anspruch umfassend einzulösen und die Unterschiede zur Softwareentwicklung einzuebnen. Gefordert ist aber auch die Unternehmensführung, den Trend zur systematischen, Team-orientierten Produktentwicklung zu fördern und das Thema ganz oben auf die Agenda zu setzen. ■



Die Software-Branche setzt schon lange auf Collaboration, Vorgehensmodelle und integrierte Qualitätssicherung. Beispiele sind das V-Modell als Vorgehensmodell oder »Information Hiding« als Gestaltungsprinzip für wartungsfreundliche Software-Systeme.

sind kein Tabu mehr. Dies macht den Blick hinüber zu den Softwareentwicklern also auch für die gestandenen Ingenieurdisziplinen interessant. Ein direkter Vergleich: Im Bereich mechanischer Produkte liegen die Arbeitsergebnisse quasi zu

wareentwicklung einzuebnen. Gefordert ist aber auch die Unternehmensführung, den Trend zur systematischen, Team-orientierten Produktentwicklung zu fördern und das Thema ganz oben auf die Agenda zu setzen. ■