

# Mechatronik im Wandel begriffen

Wie beim Product Lifecycle Management durchlief das Verständnis von Mechatronik mehrere Phasen. Heute ist nicht mehr von einer „Maschine mit Software-Anteilen“, sondern eher von „Software mit Mechanikkomponenten“ die Rede. Eine geschickte Software-Engineering-Strategie hilft, die Kundenbindung zu erhöhen.

Von RAINER STETTER und THOMAS BLUM

Die Mechatronik beschäftigt sich interdisziplinär mit dem Zusammenwirken mechanischer, elektronischer und informationstechnischer Elemente und Module in mechatronischen Systemen. Mechatronik als Schlagwort wird seit vielen Jahren in der Fachpresse intensiv diskutiert. Jedoch haben immer wieder neu aufkommende technologische Herausforderungen verdeutlicht, dass Chancen und Risiken für den Maschinenbau in der Mechatronik nach wie vor zu wenig Beachtung finden. Insgesamt sprechen wir von drei mechatronischen Wellen.

**Die Modularisierung der Mechanik (1969 bis 1985).** Wir begegnen dem Begriff „mechatronics“ 1969 erstmals in Japan (1), als versucht wurde, die erweiterten Möglichkeiten mechanischer Systeme mit Hilfe von Elektronik zu beschreiben. Anfänglich waren es einfache mechanische Systeme, die durch die Elektronik ergänzt wurden. Dass dies zuerst in Japan geschehen ist, ist auf das dort bereits damals vorhandene Know-how in der Entwicklung elektronischer Komponenten zurückzuführen. Man beherrschte die Elektronik und

suchte nach Möglichkeiten, diese Ideen in technischen Lösungen umzusetzen.

**Der Einzug des elektronischen Getriebes (1986 bis 2004).** Aufgrund der rasanten Entwicklung in der Steuerungs- und Antriebstechnik konnten in einer zweiten Welle der Mechatronik mechanische Systeme entkoppelt, modularisiert und durch ein „elektronisches Getriebe“ je nach Anwendungswunsch wieder miteinander synchronisiert werden. Die in den 90er Jahren aufkommenden Hochleistungsprozessoren erlaubten es, dass in Echtzeit immer schnellere Prozesse „mechatronisiert“ werden konnten.

Übrigens war einer der Gründe, warum der konventionelle deutsche Maschinenbau Anfang der 1990er Jahre in eine erste tiefe Krise rutsche, die Reduzierung der mechanischen Komponenten in den Systemen aufgrund der Ausbreitung mechatronischer Entwicklungsstrategien. So konnten ganze Getriebeanordnungen durch dezentrale Antriebslösungen und schnelle elektrische Synchronisation flexibel und kostensparend in Produktionsmaschinen ersetzt werden.

Diese neuen Möglichkeiten führten

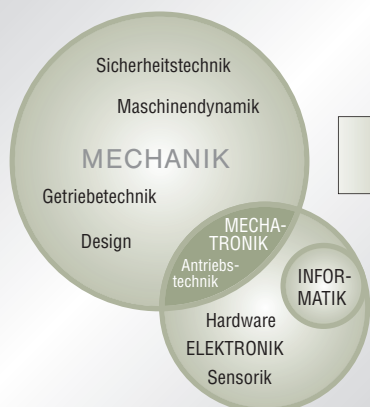
dazu, dass die physikalischen Grenzen, die zuvor durch die bewegten Massen der Mechanik bestimmt waren, überwunden werden konnten. Produktionssteigerungen durch die Modularisierung der Prozesse bis an die Belastungsgrenzen der verwendeten Materialien, etwa in der Umformtechnik, wurden möglich.

**Aufkommen des Software Engineerings (2004 bis heute).** Inzwischen hat sich deutlich gezeigt, dass die Konstruktions- und Entwicklungsprozesse von mechatronischen Lösungen immer mehr durch eine neue Disziplin, die Softwareentwicklung, in ihren zeitlichen Abläufen und Kosten dominiert werden. Waren während der zweiten Mechatronikwelle noch die Lösungen auf die zentralen Baugruppen einer Maschine beschränkt, so ermöglicht heute die Modularisierung und der Einsatz einer Vielzahl mechatronischer Komponenten auf immer tieferen Baugruppenebenen die kostengünstige, flexible Entwicklung im Maschinenbau. Viele mechatronisch geprägte Ideen stellen somit die Grundlage für die wirtschaftliche Lösung von komplexen Produktionsanforderungen dar.

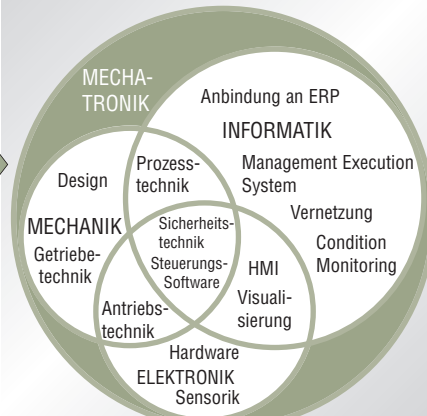
Dies führt aber dazu, dass dem Entwickeln von Software-Programmen die zentrale Bedeutung im Entstehungsprozess einer neuen Maschine zukommt. Darüber hinaus steigen die Anforderungen vom „reinen Programmieren“ hin zum zielgerichteten Engineering des gesamten Softwareentwicklungsprozesses und des Maschinenlebenszyklus an. Software Engineering findet allerdings im konventionellen Maschinenbau noch sehr zögerlich statt. Klar definierte Abläufe, wie wir sie beispielsweise aus der spangebenden Fertigung kennen, sind bis dato kaum anzutreffen. Um jedoch bei steigender Modularität und Wiederverwendbarkeit von Komponenten und Baugruppen die geforderte Flexibilität – insbesondere in der Software – in der zur Verfügung stehenden Zeit zu erreichen, bedarf es klar gegliederter, aufeinander abgestimmter Engineering-Prozesse, bei denen die Disziplin „Software“ eine wesentliche Rolle spielt.

In Zusammenarbeit mit dem VDMA wurde unter dem Stichwort „BEST-VOR – bestes Vorgehen im Maschinen- und Anlagenbau“ (2) ein Assessment Tool entwickelt, das in wenigen Stun-

Maschinenbau gestern und...  
Mechatronik, eine Teilaufgabe von



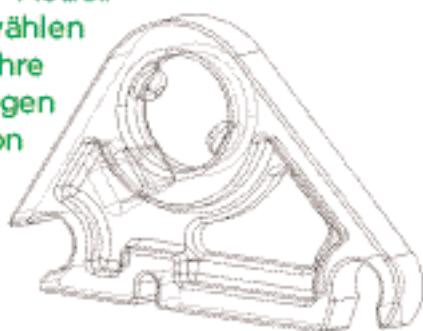
Maschinenbau von morgen!  
Mechatronik, die ganzheitliche Aufgabe.



Quelle: ITO 2009

# Echte Teile. Echt schnell.

Laden Sie einfach  
Ihr 3D-CAD-Modell  
hoch und wählen  
Sie die für Ihre  
Anforderungen  
beste Option  
aus.



## protomold®

Spritzgießen in 1-15 Tagen.  
Ideal für 10->10.000 Teile.  
Preise ab €1,495.

Wählen Sie aus  
hundertern von  
technischen  
Kunststoffen aus,  
darunter HDPE,  
Polypropylen,  
ABS/PC, Acetal,  
PBT, Polycarbonat,  
Nylon 66,  
Polyamid und  
LPDE.



## first cut®

CNC-Fräsen in 1-3 Tagen.  
Ideal für 1-10 Teile.  
Preise ab €70.

Wählen Sie aus  
30 verschiedenen  
Materialien aus,  
darunter ABS,  
Nylon, PC, Delrin,  
PEEK, ULTEM und  
Aluminium.



## Das Arbeiten mit Proto Labs ist ganz einfach.

Das Arbeiten mit Proto Labs ist ganz einfach. Sie laden Ihr 3D-CAD-Modell hoch und wählen die für Ihre Anforderungen beste Option aus. Wir kümmern uns um den Rest. Von der Materialauswahl bis zur Lieferung Ihrer Teile. Das ist Proto Labs. Einfach. Schnell. Günstig.

©2010 Proto Labs, Ltd. nach ISO 9001:2008 zertifiziert

**proto labs**  
Real Parts. Really Fast.™

Besuchen Sie noch heute [www.protolabs.de/parts](http://www.protolabs.de/parts)  
um Ihr KOSTENLOSES Exemplar unseres umfassenden  
Vergleichs der Rapid-Prototyping-Verfahren zu  
erhalten. Geben Sie den Quellcode EE10 ein.

Rufen Sie uns unter +49 (0) 6261 6741 768  
an oder besuchen Sie [protolabs.de](http://protolabs.de)

den auf effiziente, einfache Art und Weise einen Leistungsvergleich der Unternehmen erlaubt. Häufig zeigt sich bei einer ersten Evaluierung der Prozessreife, dass Maschinenbauunternehmen erst am Anfang der Einführung von durchgängigen mechatronischen Engineering-Prozessen stehen. Grund für den zögerlichen Einstieg in ein konsequentes Software Engineering ist ein damit verbundener zwingend notwendiger Umdenkprozess. Noch entwickelt man lieber „Maschinen mit Software“ und nicht „Software mit Maschinen“, was künftig verstärkt notwendig sein

wird. Denn nur so ist es vorstellbar, dass die immer größer werdende Komplexität von

- Steuerungssoftware
  - SPS-Software
  - Bedienersoftware
  - Software zur Fernwartung
  - Software zur Vernetzung
  - Software zur Prozessüberwachung
- um nur einige Facetten der Softwareentwicklung zu nennen – sicher beherrscht werden kann.

Der Lohn eines derart tiefgreifenden Strukturwandels und Umdenkprozesses im Maschinen- und Anlagenbau wird

vielfältig sein. Als Beispiel dafür kann der verstärkte Schutz vor der Produktpiraterie dienen. Der beispiellose Erfolg des deutschen Maschinenbaus im internationalen Geschäft hat viele Nachahmer auf den Plan gerufen. Es gibt sicherlich viele Anforderungen, die bei der Entwicklung und Herstellung von Mechanik zu beachten sind. Seien es Materialauswahl und -qualität, Oberflächeneigenschaften, Fertigungstoleranzen oder der Produktionsprozess an sich – in der Regel ist die Lösung am Produkt mehr oder weniger leicht erkennbar und mit etwas Erfahrung ein

## Studie: Leitfaden für SPS-Applikationen

Die Hochschule Esslingen und die Universität Duisburg-Essen sowie der Münchener Mechatronikspezialist ITQ GmbH erarbeiten zurzeit die Grundlagen für ein Benchmark-Verfahren von speicherprogrammierbaren Steuerungen.

Ziel des Projekts ist es, einen Leitfaden zu entwickeln, der ganz allgemein beschreibt, wie für eine bestimmte Applikation eine geeignete Steuerung gefunden werden kann und welche Kriterien für eine solche Auswahl wichtig sind. Hierzu wurden auf der SPS/IPC/Drives Ende November 2009 Interviews mit 14 renommierten Steuerungsherstellern geführt. Die Befragung brachte unter anderem die wesentliche Erkenntnis zutage, dass einerseits die Betrachtung der Steuerungshardware an Bedeutung verliert und dass andererseits Software-Engineering und die Programmerstellung zunehmend wichtiger werden.

**Ausgangssituation.** Die Auswahl von Steuerungen wird gerade in der heutigen Zeit im Wesentlichen am Faktor „Kosten“ festgemacht. Eine solche Sichtweise ist jedoch nicht ausreichend, da eine wirkliche Kostenoptimierung auch die Betrachtung des Anwendungsbereichs beziehungsweise des benötigten Funktionsumfangs erforderlich macht. Die Analyse des nötigen Funktionsumfangs kommt jedoch heute noch vielfach zu kurz. Auch wird die Performance der jeweiligen Entwicklungsumgebung häufig nicht ausreichend betrachtet.

**Ergebnisse der Befragung.** Die Steuerungshersteller stimmen grundsätzlich

überein, dass folgende Punkte wichtig sind, um die Zukunftssicherheit ihrer Systeme zu gewährleisten:

- Unterstützung aktueller Bussysteme
- Onboard-Diagnose
- Fernwartung
- lange Verfügbarkeit von Ersatzteilen
- weltweiter Support.

Darüber hinaus kam die Wichtigkeit der Anwendbarkeit, Skalierbarkeit und Flexibilität von Steuerungen zutage. Als unwichtiges Kriterium wurde einstimmig der Energiebedarf von Steuerungen genannt. Die Energieeffizienz spielt jedoch in der Antriebstechnik und im Aktor-/Sensor-Bereich eine immer wichtigere Rolle.

Als wichtigste Punkte für die Entwicklungsumgebung der Steuerungssysteme wurden folgende Punkte genannt:

- intuitive Bedienung
- zentrale Projektverwaltung
- umfangreiche Debugging-Möglichkeiten
- Simulation des Programms
- integrierte Testumgebung.

Zusätzlich wurden Objektorientierung, Wiederverwendbarkeit und die Unterstützung von fremden Applikationen wiederholt erwähnt.

Die genannten Punkte erleichtern die effiziente Erstellung von Anwenderprogrammen sowie das gemeinsame, gleichzeitige Arbeiten am selben Projekt. Als entscheidend wurden auch umfangreiche Testmöglichkeiten für den Entwickler genannt, die erstellten Programmbausteine unmittelbar testen zu können.

Während vor 15 Jahren die verschiedenen Steuerungssysteme noch deutlich

in der Hardware differierten, herrscht heute weitgehend Einigkeit darüber, dass sich die Hardware der meisten Hersteller nur noch wenig voneinander unterscheidet. Einige Hersteller versuchen sich dennoch über die Hardware zu differenzieren, indem sie Steuerungssysteme beispielsweise für den Einsatz bei sehr hohen und niedrigen Temperaturen anbieten.

Die Entwicklungsumgebung sowie die Unterstützung bei der effizienten Entwicklung von Software stellen heute schon wesentliche Differenzierungen dar. Somit kann der Faktor „Software“ sowie deren Engineering ein entscheidender Wettbewerbsvorteil sein.

Auch bestätigen fast alle Hersteller, dass eine objektorientierte Vorgehensweise immer wichtiger werde. Dies bedeute jedoch nicht zwangsläufig die Verwendung einer Hochsprache. Durch entsprechende Oberflächen und der Bereitstellung von verschiedenen Programmiersprachen (IEC 61131-3) in der Entwicklungsumgebung würde die Programmierung der Steuerungssysteme nicht zwangsläufig komplizierter.

**Schlussfolgerung.** Die Interviews mit den 14 Steuerungsherstellern ergaben folgendes Bild: Die reine Hardware von Steuerungssystemen ist nicht mehr der wichtigste Faktor. Eine moderne und intuitive bedienbare Entwicklungsumgebung, mit der sich effizient Anwenderprogramme entwickeln und testen lassen, gewinnt zunehmend an Bedeutung.

[www.itq.de](http://www.itq.de)

Kopieren möglich. Anders ist der Fall bei der Antriebs- und Steuerungstechnik gelagert. Hier muss sich der heimische Maschinenbau an fernöstlicher Technologie messen lassen. Im Software Engineering indes braucht man den Vergleich mit der Konkurrenz nicht zu scheuen.

Die Diskussionen über Produktpiraterie in jüngster Zeit haben immer wieder deutlich gemacht, dass jede einzelne Disziplin für sich relativ kopierbar ist, in deren Kombination das Original aber nicht nachgemacht werden kann. Mögen Ähnlichkeiten auf den ersten Blick noch den Anschein von hoher Qualität des Plagiats erwecken, so fallen bei genauerem Hinsehen die Mängel sehr schnell auf.

In der jetzigen Wirtschaftskrise zeigt sich, dass die Unternehmen es nicht zufriedenstellend geschafft haben, international eine ausreichende Kundenbindung so aufzubauen, dass „Made in Germany“ als nachhaltiges Alleinstellungsmerkmal reicht. Dies ist sicherlich ein sehr hoher Anspruch, aber die Mechatronik mit ihren Möglichkeiten könnte hierfür der Schlüssel zu einem erneuten internationalen Erfolg im Maschinenbau werden. Mit Hilfe von Software Engineering und einer entsprechenden Produkteinführungsstrategie ist es möglich, die Kundenbindung auf viele Jahre zu sichern.

So könnte in einem ersten Schritt eine Maschine nur die wesentlichen Software-Bausteine für die Steuerung der Funktionen der Maschine enthalten. Aufbauend auf dieser Basisausführung, in Verbindung mit der Mechanik und der entsprechenden Steuerungstechnik, kann dann in einem zweiten Schritt die Simulation des Prozessablaufes dem Originalprodukt hinzugefügt werden und so zu Produktionsoptimierungen und Kostenvorteilen führen. Hierbei spielt zum Beispiel die Reduzierung von Umrüstzeiten eine Rolle. Dadurch kommt es zu einer ersten Kundenbindung, trotz günstigerem Wettbewerbsangebot. Möglichkeiten der wirkungsvollen Fernwartung von Produktionsmaschinen minimieren den Nachteil gegenüber Wettbewerbern vor Ort. Die softwaretechnische Anbindung an das Intranet des Anwenders und in das WAN standortübergreifender Produktionsstätten minimieren die Nachteile weiter. Mit einem weiteren Software-Release lässt sich die Kundenbindung auch für zukünftige Investitionen weiter sichern.

Globale Angebots- und Produktionsstrategien mit vernetzten Systemen gestatten es mit Hilfe von Mechatronik, die Wettbewerbsvorteile des Originals auch bei höheren anfänglichen Investitionskosten zur rechtfertigen. Die Entscheidung für das Original ist somit strategische Grundlage für die zukünftige Unternehmensentwicklung.

Ist das Original, basierend auf einer durchdachten und zukunftsorientierten Software-Plattform, entsprechend konzipiert, kann es nur vom Hersteller kontinuierlich erweitert werden. Ohne einen Quellcode ist dies für einen Produktpiraten nicht möglich.

### Fazit

Betrachtet man den Umfang an Entwicklungsarbeiten, die mittlerweile für ein neues Produkt im Maschinenbau notwendig sind, wird deutlich, dass mittelständische Unternehmen diese Leistungen innerhalb einer akzeptablen Entwicklungszeit kaum mehr erbringen können. Denn es werden

- Kenntnisse über den Produktionsprozess des Endkunden
- Werkzeug-Know-how
- Konstruktions-Know-how in der Mechanik, Steuerungs- und Antriebstechnik
- Wissen über Prozessmonitoring
- Netzwerk-Know-how

und vieles mehr gefordert. Vertrauensvolle Kooperationen unterschiedlicher Know-how-Träger zur Lösung komplexer Maschinenbauprobleme sind erforderlich, damit der zukünftige Erfolg gewährleistet ist.

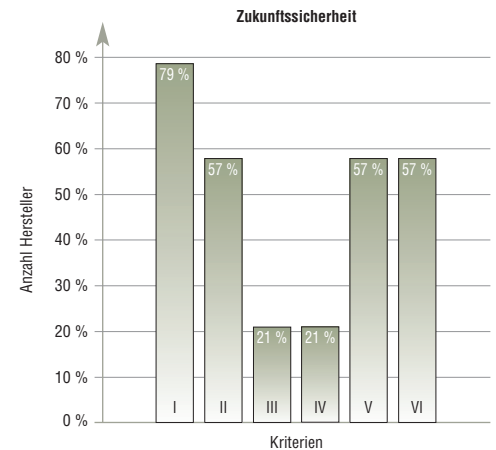
Über allem jedoch steht die Forderung, dass die Entwicklungsprozesse im Maschinenbau neu überdacht werden müssen. Software Engineering, das mit einer fundierten Initialphase beginnt, bevor durch die mechanische Konstruktion der konventionelle Entwicklungsweg bereits vorgegeben ist, ist die Voraussetzung, unter der die Chancen der dritten Mechatronik-Welle effektiv genutzt werden können.

### INFOCORNER

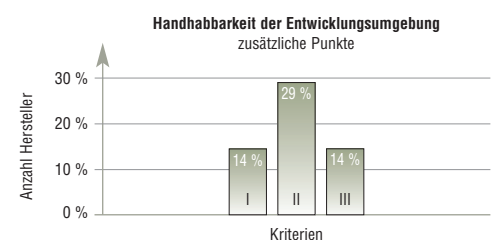
(1) Harashima, F., Tomizuka, M., Fukuda, T.: „Mechatronics – What is it, why and how?“, IEEE/ASME Transactions of Mechatronics, Vol. 1, 1996, S.1-4

(2) www.bestvor.de

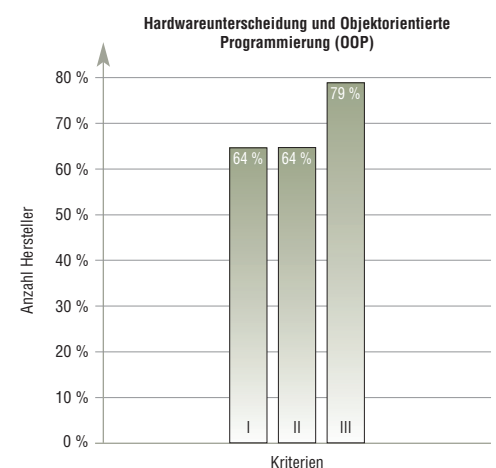
## Auswertung SPS-Benchmark (14 Hersteller)



- I = Energiebedarf nicht wichtig (jedoch in Aktorik/Sensorik wichtig)
- II = Anwendbarkeit / Handhabbarkeit wichtig
- III = Skalierbarkeit wichtig
- IV = Flexibel / Individuell wichtig
- V = Onboard-Diagnose wichtig
- VI = Bussysteme wichtig



- I = Objektorientierung
- II = Wiederverwendbarkeit
- III = Unterstützung fremder Tools



- I = kaum Unterscheidung in der Hardware
- II = Objektorientierung im Kommen
- III = OOP bedeutet nicht zwingend Hochsprache => einf. Handhabung möglich, z.B. mit FBS