

Software-Quality

Qualität beginnt im Kopf!

Autor: Dr. Rainer Stetter

„Die Maschine ist fertig, jetzt fehlt nur noch die Software.“ Dieses Zitat beschreibt einerseits die innere Haltung, andererseits aber auch den Umstand, dass man Software nicht sehen bzw. nicht berühren kann. Welchen zentralen Stellenwert das Thema Software für den Maschinen- und Anlagenbau hat, wird bei näherer Betrachtung der technischen Strukturen klar.

Eine Maschine bzw. Anlage besteht heutzutage nicht nur aus einer Vielzahl von elektronischen und mechanischen Elementen, sondern auch aus einer bunten Mixtur aus verschiedenartigen Softwarekomponenten.

Während bei den Hardwarekomponenten (Elektrik, Elektronik, Mechanik) sehr viel Wert auf Qualität gelegt wird und dem Thema Qualitätssicherung eine entsprechend große Bedeutung beigemessen wird, ist im Bereich der Softwareentwicklung das Thema Qualitätssicherung bei weitem nicht so verbreitet. Die Vernachlässigung der Software hinsichtlich Qualitätsaspekte hat oft weit reichende Folgen, denn erst durch das harmonische Zusammenspiel der einzelnen Komponenten entsteht ein Gesamtsystem, das den meist harschen Anforderungen bzgl. Verfügbarkeit Genüge leisten kann.

Die Missachtung der Software ist umso verwunderlicher, wenn man bedenkt, dass die Anwendungsgebiete der Software im Maschinen- und Anlagenbau sehr weit gefächert sind. Üblicherweise werden die Maschinen über eine Benutzerschnittstelle bedient, die in hohem Maße Software beinhaltet. Zur Verwaltung und Beschickung der Maschinen- und Produktionsdaten werden sehr häufig Datenbanksysteme eingesetzt. Die Maschinen selbst sind untereinander über entsprechende Netzwerksysteme miteinander verbunden und an die Leitstände gekoppelt. Ebenso ist Software unerlässlich zur Diagnose von Störfällen. Für das Thema Fernwartung bzw. Teleservice gilt dasselbe.

Vordringliche Aktion - embedded systems

Aber nicht nur um die Maschine „herum“, sondern auch in der Steuerung, dem Herzstück jeder Maschine hält die Software in Form der so genannten embedded software, Stück um Stück Einzug. Dass im Bereich der Steuerungstechnik ein Höchstmaß an Qualität und Sicherheit erforderlich ist, wird klar, wenn man beispielsweise bedenkt, dass ein Airbag in hohen Stückzahlen auf vollautomatisierten Fertigungsanlagen hergestellt wird. Die hohen Sicherheitsanforderungen, die an den Airbag selbst gestellt werden, pflanzen sich natürlich auf



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.

die Produktionsanlagen und somit auf die Software fort. Mit dem Begriff Prozess ist in diesem Zusammenhang nicht nur der Herstellungsprozess des zu produzierenden Gutes, sondern insbesondere der Entwicklungsprozess gemeint, der in hohem Maße über die Qualität der Anlage und somit indirekt über Güte des Endproduktes entscheidet. Denn nur ein sicherer Entwicklungsprozess garantiert ein sicheres Produkt.

Während für die klassischen Software-Anwendungsgebiete solide Vorgehensweisen für die Softwareentwicklung bereits bekannt sind - oftmals aber auch nicht angewendet werden - bleiben für den Bereich der embedded software noch viele Fragen offen. Diesen Umstand anzugehen, wurde letztes Jahr eine so genannte „Vordringliche Aktion embedded systems“ gegründet. Mit Hilfe dieses Vorhabens wurde unter Mitwirkung des Fachverbands Software des VDMAS der Handlungsbedarf für Forschung und Industrie ermittelt.

Wesentliches Element der Vordringlichen Aktion war die Befragung einer Vielzahl von Unternehmen hinsichtlich deren Vorgehensweisen und Methoden in der Softwareentwicklung. Dabei wurde bei einer nicht vernachlässigbaren Anzahl von Unternehmen deutliche Schwächen hinsichtlich deren Fähigkeiten und Fertigkeiten bezüglich Software deutlich. Die Ergebnisse spiegeln sehr gut andere Umfrageergebnisse wider. Interessant ist darüber hinaus, dass auch auf internationalem Niveau, die Qualität der Softwareerstellung nicht den Reifegrad hat wie traditionelle Ingenieursdisziplinen.

Dies liegt sicherlich zum einen in der Ausbildung, da bis heute noch in vielen Ingenieursgängen das Thema Software nur unzureichend behandelt bzw. gelehrt wird. Zum anderen ist zu erkennen, dass heute in vielen Unternehmen an verantwortlichen Stellen hinsichtlich Software, deren Bedeutung sowie den damit verbundenen Chancen und Risiken nur oberflächliches Wissen vorhanden ist. Entsprechend der daraus resultierenden Nicht- oder Minderachtung des Themas Software werden sehr häufig von ganz oben herab falsche Entscheidungen getroffen. Der Versuch, die mangelnde Sachkenntnis durch energisches Durchsetzen der (falschen) Entscheidung zu kompensieren, endet häufig tragisch und oft sehr verlustreich.

Bezüglich der finanziellen Auswirkungen fehlgeschlagener Projekte liegen keine offiziellen bzw. repräsentativen Zahlen vor, weil nur selten und wenn dann nur hinter vorgehaltener Hand über Misserfolge gesprochen wird. Jedoch kann man sich ohne große Fantasie vorstellen, dass bei Großprojekten, deren Gesamtvolumen sich oftmals auch in zwei- oder dreistelliger Millionenhöhe bewegt, Verluste ent



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.

stehen können, die gerade in diesem Umfeld häufig tätige, mittelständische Unternehmen schwer treffen.

Typischerweise ist bei solchen Projekten die Situation anzutreffen, dass die Spezifikation des Liefer- und Leistungsumfangs nur sehr schmal ausgelegt ist. Als einzig wirklich messbare Kriterien, dienen meist die Verfügbarkeit bzw. der Durchsatz. Dabei ist zu beachten, dass der Durchsatz wiederum sehr eng mit der Verfügbarkeit verknüpft ist. In der Regel hängen zu leistende Zahlungen vom Erreichen zugesicherter Leistungsdaten ab. Somit entscheidet die Qualität letztlich über Liquidität, Umsatz und Erlös.

Reifegradmodell CMM (Capability Maturity Model)

Trotz dieser an für sich einfachen Zusammenhänge sind die Unternehmen in der Minderzahl, die systematisch an der Verbesserung ihrer Entwicklungsprozesse arbeiten. Diese Behauptung wird durch Studien untermauert, die - wie sollte es im Softwarebereich anders sein - ursprünglich nur in den USA durchgeführt wurden. Zur Bewertung der Prozessgüte wurde ein so genanntes Reifegradmodell entwickelt. Mindestens zwei Drittel der befragten Unternehmungen befinden sich auf dem so genannten Initial- Status, der dadurch charakterisiert wird, dass selbst viele fundamentale Kernpraktiken, wie beispielsweise eine strukturierte Vorgehensweise für Test- und Qualitätssicherung der Software, nicht vorhanden sind.

Das besondere an dieser Statistik ist, dass in den USA Softwareunternehmen nach diesem Schema bewertet werden, die für staatliche Organisationen insbesondere für militärische Bereiche tätig sind. In diesem Zusammenhang erschreckt dieses Ergebnis umso mehr, da diese Bereiche in jedem Fall sicherheitskritische Aspekte beinhalten.

Projiziert man diese Erkenntnisse auf die Situation im deutschen Maschinen- und Anlagenbau ergibt sich sicherlich kein geringerer Handlungsbedarf. Vielmehr weisen die Umstände darauf hin, dass hier noch mehr Potential offen liegt. Dies liegt insbesondere darin, dass die sich stellende, interdisziplinäre Problemstellung noch weit schwerer zu lösen ist als eine reine Softwareentwicklung. Eine interdisziplinäre Entwicklung hat aber nicht nur damit zu tun, dass verschiedene Technologien zusammengeführt werden. Viel schwerer wiegt, dass Entwicklungsgruppen mit völlig unterschiedlichen Traditionen und Denkweisen geradezu aufeinander prallen. Während Hardwareentwickler daran gewöhnt sind nach Normen und Richtlinien zu arbeiten, verstehen sich viele Softwareentwickler als



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.

Künstler und Artisten, deren Arbeit selbstverständlich in kein reglementierendes Korsett gesteckt werden kann. Sehr häufig ist zudem zu beobachten, dass die einzelnen Fakultäten (Mechanik, Elektronik und Software) in verschiedenen Organisationseinheiten disziplinarisch eingeordnet sind. Die damit verbundene Abgrenzung zwischen den Entwicklungseinheiten führt dabei gerne dazu, dass die Gesamtproblemstellung jeweils nur unter ganz speziellen Gesichtspunkten gesehen wird und viele Probleme erst in der Integrationsphase, also sehr spät im Projekt entdeckt werden.

Meist ist auch zu beobachten, dass die Software nur als Anhängsel oder notwendiges Übel gesehen wird.

Besonders prägnant wird die Meinung hinsichtlich Software und deren Bedeutung durch den oftmals zu vernehmenden Satz „Die Maschine ist fertig, jetzt fehlt nur noch die Software“ charakterisiert. Dieses Zitat beschreibt einerseits die innere Haltung, andererseits aber auch den Umstand, dass man Software nicht sehen bzw. berühren kann. Das einzige was man in der Regel von Software weiß, ist, dass sie nicht funktioniert. Software und deren Strukturen sichtbar zu machen, ist sehr schwierig. Deshalb erfordert die Software ein sehr hohes Abstraktionsvermögen.

Notwendigkeit einer interdisziplinären Betrachtungsweise

Ein weiterer Aspekt der beachtet werden muss, ist der unterschiedliche Reifegrad der einzelnen Ingenieursdisziplinen. Während die Prozesse zur Entwicklung von mechanischen und elektronischen Komponenten einen sehr hohen Reifegrad erreicht haben, sind in der Softwareentwicklung die Vorgehensweisen weitaus weniger ausgearbeitet. Nachdem die einzelnen Komponenten zu einem Gesamtsystem integriert werden müssen, werden quasi die Reifegrade der verschiedenen Entwicklungseinheiten addiert. Dabei ist aber zu beachten, dass durch diese Quasi-Addition die Gesamtreife des Prozesses dramatisch abnimmt.

Die Arbeitsabläufe sind heute meist so gestaltet, dass im Projektverlauf zunächst die Hardware betrachtet wird. Sollten in dieser Phase Problempunkte nicht erkannt oder übersehen, was bei der Lösung komplexer Anforderungen durchaus der Fall sein kann, muss die Software diese „geradebiegen“.



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.

Die technischen Strukturen heutiger Automatisierungslösungen bedürfen aber gleichzeitiger und gleichwertiger Betrachtung aller Disziplinen. Daraus folgt, dass die Arbeitsabläufe völlig neu strukturiert werden müssen. Dies zieht wiederum das Durchschreiten einer Lernkurve nach sich. Wie jeder aus eigener Erfahrung weiß, braucht man um zu lernen eine gewisse Muse und überhaupt die Einsicht, dass es sich lohnt etwas zu lernen.

Für den Maschinen- und Anlagenbau würde es sich sicher lohnen, sich mit dem Software mehr auseinander zu setzen, denn durch eine bessere Integration der Software in die Systeme würden sich hohe Rationalisierungspotentiale ergeben. Einerseits könnten bei Verzug zu leistende Konventionalstrafen, andererseits könnten Minderzahlungen und Kürzungen bei nicht erreichten Leistungsdaten verhindert werden. Weiterhin könnten aber durch einen systematischen und gut durchdachten Einsatz von Software völlig neue Ideen realisiert werden. Die daraus resultierende Innovationskraft wäre von entscheidender Bedeutung, um auch mittelfristig auf den internationalen Märkten zu bestehen. Wenn die in der Software steckenden Chancen und Risiken nicht rechtzeitig erkannt werden, kann es leicht sein, dass Deutschland wie in so vielen anderen Hochinnovationsmärkten mittelfristig das Hintertreffen hat.

Sichtbarmachung der Software

Nachdem die Auswirkungen von verfehlter Bildungspolitik und fehlendem Verständnis und damit verbundenen mangelndem Bewusstsein im Management nicht ohne weiteres und schwerlich von jetzt auf morgen zu überwinden sind, muss man in überschaubaren, aber zielstrebigem Schritten daran arbeiten, die Strukturen und Arbeitsabläufe an die neue Problemstellung anzupassen. Ein erster Schritt der sichtbar machen würde, dass man die Software ernst nimmt, wäre sich intensiv mit dem Thema Software-Qualitätssicherung auseinander zu setzen. Dass zur Messung der Softwarequalität weder Schiebellehre noch Oszilloskop taugen, ist klar. Trotzdem ist es nicht so, dass es keine Verfahren und Ansätze gäbe, um Softwarequalität sichtbar zu machen. Diese erfordern jedoch eine völlig neue Arbeitsweise und vor allem Denkweise in der Industrie.

Zur Sichtbarmachung der Software bedarf es keiner Hexerei, sondern lediglich der Anwendung bzw. Übertragung von den in traditionellen Ingenieursdisziplinen bekannten Vorgehensweisen auf das Softwareengineering. Was zu tun wäre und wie groß hier der Handlungsbedarf ist, soll an verschiedenen Beispielen erläutert werden.



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.

Keinem Hardwarehersteller würde einfallen, seine Einzelteile, Komponenten, Baugruppen etc. irgendwo im Lager, ohne Nummerierung, ohne System aufzubewahren. Heutzutage sind überall in der Industrie mehr oder weniger ausgefeilte Systeme vorhanden, um eine solide Stücklistenverwaltung zu gewährleisten. Hingegen ist zu beobachten, dass die Verwaltung der „Softwarestücke“ sehr häufig noch in den Kinderschuhen steckt.

Verschiedene Umfragen haben ergeben, dass in ca. 70% der untersuchten Unternehmen keine professionellen Hilfsmittel eingesetzt werden, um die „Softwarestücke“ zu verwalten. Vielmehr ist die Situation anzutreffen, dass die Quelldateien, d.h. die „softwaretechnischen“ Einzelteile, mehr oder weniger unstrukturiert und undokumentiert auf verschiedenen Festplatten einzelner Entwickler gespeichert sind. Das hat sehr häufig zur Folge, dass falsche Versionen von einzelnen Dateien zusammengebunden werden und sich die lauffähige Software auf einmal ganz anders verhält als vermutet. Ein daraus resultierender oft zu beobachtender „netter“ Effekt ist, dass ein in einer Version bereits behobener Softwarefehler in der nächsten Softwareversion „plötzlich“ wieder auftaucht, weil die falschen Dateien zusammengebunden wurden.

Sinnvolle Softwaretests

Nachdem in einem Unternehmen ein funktionierendes Instrument zur Verwaltung der Softwarebestandteile, ein so genanntes Konfigurationsmanagementsystem installiert ist, sind erst die Voraussetzungen erfüllt, um sich mit dem Thema Qualitätssicherung sinnvoll auseinander zu setzen. Denn nur wenn klar ist, aus welchen Einzelteilen eine Softwareversion besteht, macht ein Test überhaupt erst Sinn.

Beim Softwaretesten verhält es sich ähnlich wie bei der Verwaltung der Softwarekomponenten. Auch hier ist in verschiedenen Studien herausgearbeitet worden, dass in bis zu 75% der untersuchten Unternehmen keine sauber strukturierten und dokumentierten Softwaretests durchgeführt werden. Inwieweit diese Zahl schlüssig ist, wird unmittelbar klar, wenn man sich vor Augen führt, wie viele Unternehmen Test bzw. Prüfabteilungen besitzen, die mit dem Thema Softwaretest vertraut sind und somit auch sinnvolle Softwaretests durchführen können. Im Vergleich dazu sollte man sich überlegen, wie viel Energie im Bereich der Hardware in die Qualitätssicherung gesteckt wird. An dieser Stelle wäre wahrlich überlegenswert, ob es nicht sinnvoll wäre, einen Teil der Anstrengungen hinsichtlich Qualitätssicherung von der Hardware auf die Software zu verlegen. Denn aus unserer Erfahrung sind die Qualitätsstandard zwischen Hard- und



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.

Software meist sehr unterschiedlich. Die Qualität des Gesamtsystem Maschine/Anlage wird jedoch durch die Qualität des schwächsten Gliedes bestimmt.

Die Softwarequalitätssicherung wird aber durch einen weiteren, weit verbreiteten Missstand erschwert, teilweise sogar nahezu unmöglich gemacht. Damit etwas getestet werden kann, muss klar sein, was das zu testende Element tun soll. Die Anforderungen müssen dabei klar sein. Auch hier gibt es Studien, die besagen, dass nur in ca. 20 % der Fälle, vor Beginn der Realisierung klar ist, was überhaupt realisiert werden soll. Unter diesen Voraussetzungen ist es nicht weiter verwunderlich, dass die Software über den ganzen Projektverlauf so unsichtbar bleibt. Wie kann denn etwas transparent werden, wenn es von Beginn an ignoriert wird? Aus diesem Grunde müsste von Anfang in die Anforderungsanalyse die Software miteinbezogen werden. Zur Darstellung der Strukturen und Abläufe die die Software betreffen, gibt es inzwischen schon weit gereifte Verfahren und Werkzeuge (die z.B. auf UML beruhen), die es möglich machen, Software sichtbar zu machen.

Fazit

Die Vorgehensweisen und Verfahren zur Sichtbarmachung der Software sind zumindest soweit gediehen, dass man sie einsetzen kann. Trotzdem kommen Sie häufig nicht zum Einsatz.

Unklar bleibt warum? Ist es mangelnde Einsicht oder Unvermögen? Wahrscheinlich eine Mischung aus beidem.

© Dr. Rainer Stetter, ITQ GmbH



Unser Ziel ist es, Ihr Wissen und Ihr Know-How durch überzeugende und qualifizierte Weiterbildungen zu erweitern und Sie so in Ihrer täglichen praktischen Arbeit effektiv zu unterstützen.