Die approbierte Originalversion dieser Diplom-/Masterarbeit ist an der Hauptbibliothek der Technischen Universität Wien aufgestellt (http://www.ub.tuwien.ac.at).

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology (http://www.ub.tuwien.ac.at/englweb/).

# DIPLOMARBEIT

## Anforderungsgetriebene Szenarienevaluierung zur Modernisierung von Qualitätsystemen in der Fertigungsindustrie

zur Erlangung des akademischen Grades
Diplomingenieur
(Dipl.-Ing.)

ausgeführt am
Institut für Rechnergestützte Automation
Forschungsgruppe Industrial Software

der Technischen Universität Wien

unter der Anleitung von
Univ.-Prof. Dipl.-Ing. Dr. Thomas Grechenig

durch
Robert Messner

Rosaliagasse 9/26
1120 Wien

Wien, 09.06.2008

## Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am _____ _____

Name

## Danksagung

an alle ...

## Kurzfassung

Qualitätsicherung ist eine zentrale Säule in der Fertigungsindustrie, wobei in diesem Bereich in vielen Unternehmen in den letzten Jahren Softwarelösungen historisch gewachsen sind, die heutzutage nicht mehr State of the Art sind und einer Erneuerung unterzogen werden müssen, um auf dem Markt konkurrenzfähig bleiben zu können.

Ziel dieser Arbeit ist es aufzuzeigen, wie man Legacy Systeme identifiziert und klassifiziert, und wie man eine Strategie zur Modernisierung entwickelt, die für das Unternehmen als Ganzes eine optimale Lösung darstellt. Nach einem allgemeinen Überblick über die Grundlagen der Qualitätssicherung wird anhand eines konkreten Fallbeispiels im Rahmen der vorliegenden Arbeit untersucht, welche Maßnahmen zur Modernisierung des bestehenden, Software-basierten Qualitätsystems am optimalsten geeignet sind.

Dabei wird anhand einer Szenario-basierten Entscheidungsfindung und eines Reverse-Engineerings eine vergleichende Analyse durchgeführt, die Vor- und Nachteile der unterschiedlichen Strategien aufzeigt.

## Abstract

Quality assurance plays a central role in the manufacturing industry. Many companies have accumulated solutions in this area, which have been growing historically. These solutions are usually not competitive and "state of the art" any more. In order to stay competitive, these solutions have to be revised. The goal of this work is to point out how to identify and classify legacy systems, and how to implement a modernization strategy, which assures an optimal solution for the company.

After giving a general overview of the basic principles of quality assurance, a concrete example is depicted in great detail. In this example it will be analyzed which actions need to be taken in order to modernize and optimize the existing software based quality assurance system.

Scenario based decision making and reverse engineering is used to make a comparative analysis, which shows the advantages and disadvantages of the different possible strategies.

# Inhaltsverzeichnis

1 Einleitung.....12

2 Grundlagen der Qualitätssicherung.....20
2.1 Begriff und Definition von CAQ.....20
2.2 Module eines CAQ Systems.....21
2.2.1 Prüfplanung.....21
2.2.2 Wareneingangsprüfung.....23
2.2.3 Warenausgangsprüfung / Endprüfung.....24
2.2.4 Erstmusterprüfung.....25
2.2.5 FMEA.....26
2.2.6 Fertigungsprüfung und SPC.....27
2.2.7 Reklamationsmanagement.....28
2.2.8 Prüf- und Messmittelverwaltung.....30
2.3 Qualitätswerkzeuge.....31
2.4 Modellbeschreibung des Produktionsprozesses.....34
2.4.1 Prozessmodell A.....36
2.4.2 Prozessmodell B.....37
2.4.3 Prozessmodell C.....38
2.5 Klassifizierung von Qualitätsmerkmalen.....39
2.6 Darstellung von Messergebnissen.....41
2.6.1 Darstellung von Einzelwerten.....41
2.6.2 Wertestrahl.....42
2.6.3 Histogramm.....43
2.6.4 Darstellung von statistischen Kennwerten.....44
2.7 Qualitätsregelkartentechnik.....45
2.7.1 Aufbau und Interpretation einer Regelkarte.....46
2.7.2 Beispiele Qualitätsregelkarten.....50

3 Analyse und Klassifizierung von Legacy Systems.....55
3.1 Software Wartung.....55
3.1.1 Abstraktionsgrade und -ebenen.....56
3.1.2 Definitionen.....58
3.1.3 Ziele des Reverse Engineering.....62
3.2 Legacy Systems.....65
3.3 Umgang mit Legacy Systems.....67
3.3.1 Strategien für die Zukunft.....67

3.3.2 Das SABA model....71
4 Qualitätssicherung in der GKN Sintermetals....78
4.1 GKN Unternehmensprofil....78
4.2 GKN Sintermetals ....79
4.3 Eckdaten des Unternehmens....81
4.3.1 Allgemeine Informationen....81
4.3.2 Qualitätszertifikate....82
4.3.3 Produktionsanlagen....82
4.4 Unternehmensstruktur....83
4.5 Entwicklung des Qualitätswesens....84
4.6 Ausgangsbasis und Geschichte der Qualitätssoftware....85
4.7 Systemgrenzen und technische Architektur....85
4.8 CAQ Module Ist-Erhebung....87
4.9 Verbesserungspotential und Maßnahmen....89
5 Technische Analyse des bestehenden Systems....91
5.1 Projektverlauf und -ziel....91
5.2 Reverse Engineering....92
5.2.1 Analyse des Quellcodes....94
5.2.2 Apprenticing (in die Lehre gehen)....97
5.2.3 Benutzerbefragungen....98
5.2.4 Ergebnisse des Reverse Engineering....99
5.3 Deckung der Anforderungen mit CASQ-IT 9000....106
5.3.1 Delta Analyse....106
5.3.2 Change Prozess....107
6 Evaluierung möglicher Lösungsszenarien....110
6.1 Szenario Standardsoftware....110
6.1.1 Allgemeine Szenariobeschreibung....110
6.1.2 Deltas unbedingt nötige Anforderungen....112
6.1.3 Auswirkungen nicht gedeckter Anforderungen bei Einführung von CASQ-IT 9000....117
6.1.4 Zusätzlich erforderliche Systeme....122
6.1.5 Vor- und Nachteile....123
6.1.6 Kosten....124
6.1.7 Anmerkung zu Böhme und Weihs....125
6.2 Szenario Reengineering....125

6.2.1 Allgemeine Szenariobeschreibung.....125
6.2.2 Vorgehensweise und Aufbau.....126
6.2.3 Vor- und Nachteile.....127
6.2.4 Kosten.....130
6.3 Vergleich beider Szenarien.....130
6.3.1 Hardware.....130
6.3.2 Software.....131
6.3.3 Rollout.....132
7 Zusammenfassung und Ausblick.....133

# Abbildungsverzeichnis

Abbildung 1: Aufbau der Arbeit.....16
Abbildung 2: Gut Schlecht Denken.....31
Abbildung 3: Verlust in Abhängigkeit von Lage und Streuung....33
Abbildung 4: Verbesserte Verlustabhängigkeit.....33
Abbildung 5: Veränderte Prozesslage.....34
Abbildung 6: Prozessmodell A.....36
Abbildung 7: Prozessmodell B.....37
Abbildung 8: Prozessmodell C.....38
Abbildung 9: Merkmalsarten.....39
Abbildung 10: Darstellung von Einzelwerten.....42
Abbildung 11: Wertestrahl.....42
Abbildung 12: Histogramm.....44
Abbildung 13: Leere Regelkarte.....47
Abbildung 14: Beispiel Regelkarte.....48
Abbildung 15: Fehlersammelkarte.....51
Abbildung 16: Urwertkarte.....52
Abbildung 17: X (quer) / S - Karte.....53
Abbildung 18: X (quer)/R - Karte.....54
Abbildung 19: Reverseenigeering, Begriffe und Beziehungen[15] .....58
Abbildung 20: Tools und Methoden[15].....63
Abbildung 21: TST stage in overview.....75
Abbildung 22: Organigramm Sintermetals Bruneck.....83
Abbildung 23: Vorgehensweise.....93
Abbildung 24: Use Case Diagramm Prüfplanung.....101
Abbildung 25: Use Case Diagramm SPC .....104
Abbildung 26: Aktivity Prüfplan anlegen.....105
Abbildung 27: Changeprozess.....107
Abbildung 28: Tätigkeiten im Unternehmen.....108
Abbildung 29: Aufwand in der Softwareentwicklung.....125

# Tabellenverzeichnis

Tabelle 1: Zeitachse Gkn Sintermetals.....81
Tabelle 2: Zeitachse Entwicklung des Qualitätswesens.....84
Tabelle 3: CAQ Bruneck Software Module .....88
Tabelle 4: CAQ System Bruneck Software Lieferanten.....88
Tabelle 5: Programmcode Komplexitätsanalyse.....95
Tabelle 6: Kostenaufstellung Szenario B&W.....124
Tabelle 7: Kostenabschätzung Szenario 2 Reengineering.....130
Tabelle 8: Szenarienvergleich Hardware.....131
Tabelle 9: Szenarienvergleich Software.....132
Tabelle 10: Szenarienvergleich Rollout.....132
Tabelle 11: Übersicht CAQ Anbieter.....CXLII

# Begriffsbestimmungen und Abkürzungen

| | |
|---|---|
| APQP | Advanced Product Quality Planning |
| BDE | Betriebsdatenerfassung |
| B&W | Böhme und Weihs Systemtechnik GmbH & Co KG |
| CAQ | Computer Aided Quality |
| FMEA | Failure Mode and Effects Analysis<br>Fehlermöglichkeits- und Einflussanalyse |
| MDE | Maschinendatenerfassung, Teil der BDE |
| OST | organisational szenario tool |
| QRK | Qualitätsregelkarte |
| SPC | Statistical Process Control |
| TST | technical szenario tool |

# 1 Einleitung

„Never touch a running system!" Dieser pragmatische Grundsatz bringt wohl den Zustand vieler IT Systeme auf den Punkt. Bestehende Anforderungen an die Systeme verändern sich, neue kommen hinzu, wieder andere werden nicht mehr benötigt. Jede Änderung an einem laufenden System birgt Risiken und viele Änderungen sind wirtschaftlich nicht mehr tragbar. Trotzdem müssen sie früher oder später durchgeführt werden, ansonsten wird das System unbrauchbar oder irrelevant.

Banken, Verkehrsbetriebe, öffentliche Verwaltungen – wo immer große Organisationen in irgendeiner Form ein umfassendes, flächendeckendes IT System in Verwendung haben, kommen sie irgendwann an den Punkt, an dem sie von der rasanten technologischen Entwicklung überrollt werden. Ihre bestehenden Systeme arbeiten zwar nach wie vor einwandfrei, trotzdem sind Ersatzteile oder ganze Komponenten nur noch schwer zu bekommen. Dabei würde neue, billige Hardware ebenso ihren Dienst verrichten – doch die Umstellung wäre alles andere einfach. So verrät so manche VT Terminal Emulation auf einer Windows Oberfläche, dass im Hintergrund ein „gutes altes Unix" seinen Dienst verrichtet. Aber warum fällt eine Umstellung so schwer? Warum kann man nicht einfach wie zu Hause den alten Computer samt Software dem Recycling Prozess zuführen, die Daten kopieren, einen besseren und schnelleren Rechner in Betrieb nehmen und die alte Software installieren?!

So einfach das klingt, so oft es gemacht wurde, so viele Geschichten gibt es auch, die zeigen, dass dieser Vorgang – auch für nur einen Computer – doch nicht so einfach war wie anfangs angenommen. Bei

einem komplexen System, das hunderte von Clients und eine Vielzahl von Servern beheimatet, ist eine Umstellung noch viel komplexer und jede Änderung will genau geplant sein, damit das System als Ganzes nicht gefährdet wird.

So ist diese Arbeit in einem betrieblichen Umfeld entstanden, in dem vor mehr als 20 Jahren ein System für die Qualitätssicherung eingeführt wurde. Es wurde immer größer und umfassender und schlussendlich unentbehrlich für den Betrieb. Doch nun wird von einem veralteten System gesprochen. Darum soll es durch ein neues, besseres ersetzt werden. Aber Begriffe wie neu und besser sind relativ. Und um sie im Bezug auf Qualitätsysteme richtig zu verwenden, muss zuerst die Frage erörtert werden, wozu ein derartiges System überhaupt benötigt wird. Denn eine Umstellung verschlingt nicht nur viel Zeit und Ressourcen – die Investition muss auch gerechtfertigt und für das Unternehmen rentabel sein.

Europäische Unternehmen wurden bereits in der Vergangenheit und werden vor allem in Zukunft durch die Öffnung von Osteuropa und der zunehmenden Präsenz asiatischer Unternehmen einem verschärften Wettbewerb ausgesetzt. Eine Änderung dieser Entwicklung ist nicht abzusehen. Fragen nach dem Standortvorteil und strategischen Wettbewerbsfaktoren werden gestellt. Qualitativ hochwertige Produkte können die Position am Markt stärken. Aber was ist Qualität?

Qualität ist nach ISO 8402[1] folgendermaßen definiert:

*Qualität ist die Gesamtheit der Merkmale und Merkmalswerte eines Produkts oder einer Dienstleistung bezüglich der Eignung, festgelegte und vorausgesetzte Erfordernisse zu erfüllen.*

Diese Erfordernisse werden vom Kunden, also vom Markt vorgegeben. Die Kundenwünsche gilt es hinsichtlich folgender Gesichtspunkte zu erfüllen:

- Funktion
- Sicherheit und Zuverlässigkeit
- Umweltverträglichkeit
- Lieferzeiten
- Preise
- Beratung und Betreuung

Die Sicherung all dieser Punkte ist die zentrale Aufgabe des Qualitätsmanagements. Doch Qualität deckt nur die Kontrolle der Eigenschaften der Produkte ab, der Gedanke wurde aber in den Total Quality Control Konzepten (TQM) auf das gesamte Unternehmen ausgedehnt. Die Produkte sollen einer ständigen Verbesserung unterliegen und alle Beteiligten – von den KapitalgeberInnen über die MitarbeiterInnen, und KundenInnen bis zu den LieferantInnen – sollen eingebunden werden.

TQM wurde von der japanischen Automobilindustrie in den fünfziger Jahren des vorigen Jahrhunderts entwickelt und umfasst ein durchgängiges, dokumentierendes und sich stetig verbesserndes System, das sich über das gesamte Unternehmen ausdehnt. Oberste Priorität dabei ist die Kundenzufriedenheit, und diese kann dauerhaft nur in einem sich entwickelnden Unternehmen gewährleistet werden. Der Erfolg gab dem TQM Konzept recht und in den späten 80er Jahren kam es schließlich auch nach Europa. Von einigen Großunternehmen wurde die European Foundation for Quality Management (EFQM) aus der Taufe gehoben, die die Entwicklung eines europäischen Modells für Qualitätsmanagement vorantrieb.

Heute ist ein weitreichendes Qualitätsmanagment unverzichtbarer Bestandteil jedes erfolgreichen Unternehmens, und eben diese Unternehmen können ihre Systeme nach ISO 9000, ISO TS 16494[2], etc. zertifizieren lassen.

Durch die wachsenden Anforderungen an die Produkte und die zunehmende Komplexität ist auch die Sicherung der Qualität komplexer geworden und in großen Unternehmen nur noch durch den Einsatz von Informationsystemen wirtschaftlich zu bewältigen.

Das System für die Qualitätskontrolle, das Gegenstand dieser Arbeit ist, wurde Ende der 80er Jahre, in Zusammenarbeit mit der University of Aarhus, im Zuge einer Neuorganisation des Managementsystems, entwickelt. Das System wurde über die Jahre optimiert, laufend verbessert und den wachsenden Anforderungen angepasst. So ist es an einen Punkt gelangt, an dem Änderungen nur mehr schwer durchführbar sind und jede Änderung mit großen Risiken verbunden wäre. Nun stellt sich die Frage: Ist hier ein Legacy System entstanden? Und was ist überhaupt ein Legacy System?

Die Problemstellung ist klar. Ein Softwaresystem ist in die Jahre gekommen und aufgrund vieler Faktoren besteht Handlungsbedarf. Auf solche Situationen reagieren viele Unternehmen mit einer Bauchentscheidung. So wird die Erneuerung oft ohne konkreten Plan in Angriff genommen und versucht, das Gesamtsystem irgendwie auf einen „neueren" Stand der Technik zu bringen. Dabei wird oft vergessen, dass es eine bestehende IT-Landschaft im Unternehmen gibt. So wird diese selten mit der neuen Technologie abgestimmt. Es entstehen Insellösungen und oft ist das neue System bereits bei der Einführung ein Legacy System, ein System, das mit viel zu schwachen Techologien entwickelt wurde und ähnliche Verschleißerscheinungen trägt wie das alte. Ein System also, das die Probleme des alten nicht beseitigt, sondern nur verlagert.

Ziel dieser Arbeit ist es nun, aufzuzeigen, wie man Legacy Systeme identifiziert und klassifiziert, und wie man eine Strategie entwickelt, die für das Unternehmen als Ganzes eine optimale Lösung darstellt. Des weiteren werden auch die Grundlagen der Qualitätssicherung, die

im speziellen dieses System betreffen, aufgezeigt, da das System in einem Unternehmen im Qualitätswesen eingebettet ist und hier spezielle Anforderungen gelten.

Folgende Grafik beschreibt den grundlegenden Aufbau der Arbeit:

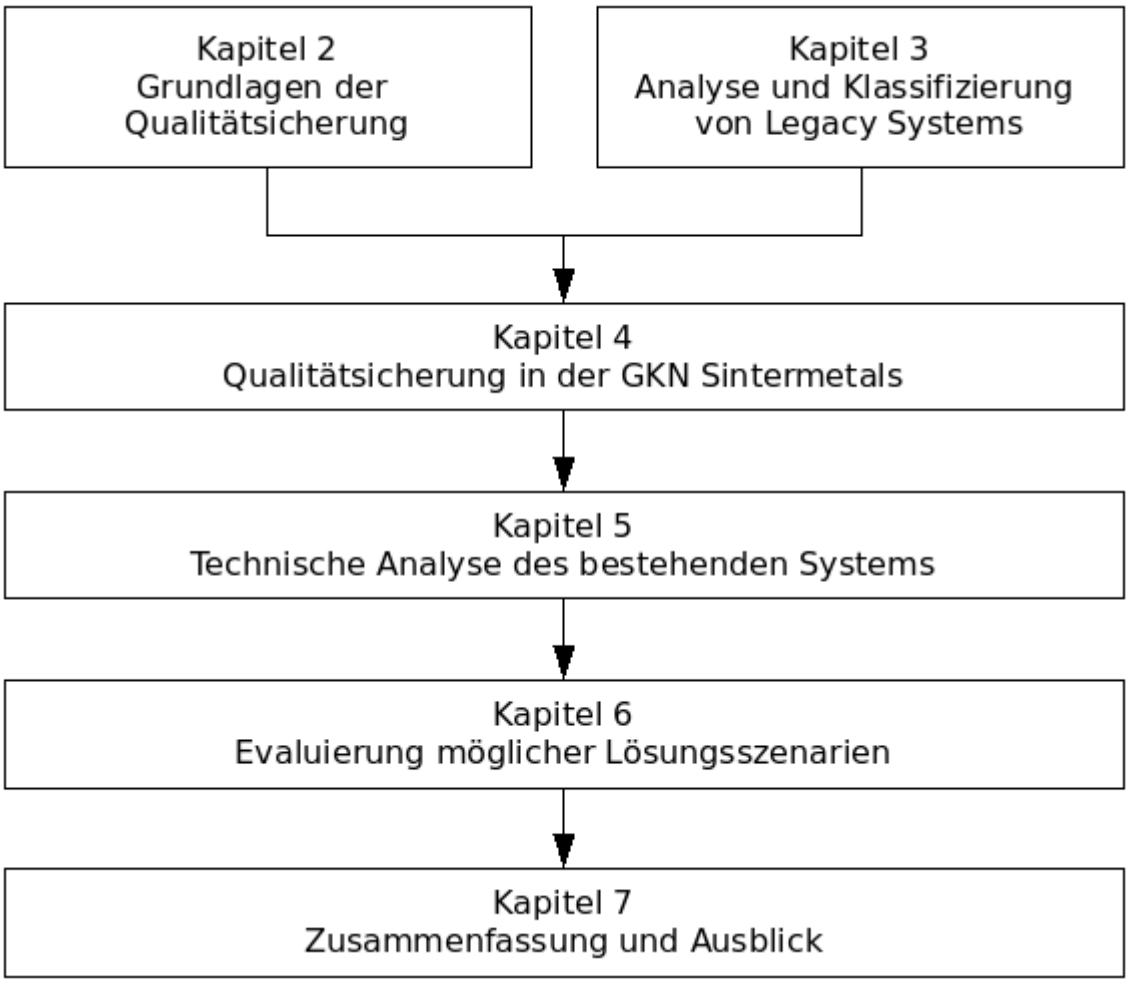


Abbildung 1: Aufbau der Arbeit

**Kapitel 2** beschreibt die Grundlagen der Qualitätssicherung, wie sie im Unternehmen durchgeführt wird. Das Qualitätswesen ist in einem modernen Unternehmen flächendeckend implementiert, Fehler sollen bereits zum Zeitpunkt ihrer Entstehung diagnostiziert und im Idealfall behoben werden. Fehler können bereits bei der Produktplanung

entstehen, beim Wareneingang, wenn qualitativ minderwertige Rohstoffe verwendet werden, in der Produktion mit ihren verschiedenen Schritten bis hin zu dem Zeitpunkt, an dem das fertige Produkt das Werk verlässt, bzw. bei den KundInnen ankommt.

In all diesen Stationen werden unterschiedliche Anforderungen an das Qualitätswesen und somit an das gesamte Qualitässystem gestellt. Dieses besteht aus verschiedenen Teilsystemen von verschiedenen Lieferanten. Ein Fokus liegt dabei auf der SPC, der Statistischen Prozess Kontrolle, sowie der Waren Ein- und Ausgangsprüfung, da diese einen wesentlichen Bestandteil des betrachteten Softwaresystems ausmacht.

In **Kapitel 3** wird dann die zweite theoretische Säule behandelt, die sogenannten Legacy Systems. Dabei wird der Software Lifecycle betrachtet, um ein besseres Verständnis für die Ursachen der Entstehung solcher Systeme zu gewährleisten. Danach werden Strategien für den Umgang mit derartigen Systemen aufgezeigt und im Speziellen eine Möglichkeit beschrieben, wie man sich durch interative Szenarienbildung den Lösungsweg erarbeitet. Dabei ist nicht nur die technische Sicht der Dinge ausschlaggebend, sondern auch die Unternehmensstrategie als Ganzes.

In **Kapitel 4** diese Theorie anhand eines Fallbeispiels aus der Praxis untermauert. Dort wird das Unternehmen im Allgemeinen, sowie das dort implementierte Qualitätswesen und seine Enstehungsgeschichte beschrieben. Ein Teil dieser Geschichte befasst sich mit der Enstehung des Qualitätssystems, das über viele Jahre entwickelt und verfeinert wurde. Weiters wird das System eingegrenzt und der Ist-Stand beschrieben. Auch wird deutlich, warum Handlungsbedarf besteht, aus welchem Grund mit dem System in dieser Art und Weise nicht mehr verfahren werden kann, wo Verbesserungspotential vorhanden ist und welche Maßnahmen gesetzt werden müssen.

Dem folgt eine technische Analyse des Systems in **Kapitel 5.** Sie teilt sich in ein Reverse Engineering, mit dem die ursprünglichen Anforderungen an das System ermittelt werden und in einen Vergleich mit einer Standardsoftware. Das Reverse Engineering baut auf drei grundlegenden Säulen:

- der Analyse des Quellcodes
- Apprenticing
- Benutzerbefragungen

Die Ergebnisse stellen die ursprünglichen Anforderungen dar, die dann mit der Standardsoftware abgeglichen wurden.

Daraus lassen sich dann in **Kapitel 6** verschiedene Szenarien bilden, im Speziellen die Einführung einer Standardsoftware oder die Durchführung eines Reengineerings. Die Szenarien zeigen mögliche Lösungswege auf, die dann verglichen werden und so eine Entscheidungsgrundlage für das weitere Vorgehen mit dem System darstellen.

Ziel der Arbeit ist es, die beste Lösungsstrategie herauszufinden und somit die weitere Vorgehensweise zu bestimmen. Für gewöhnlich sind eine Unmenge an Möglichkeiten gegeben, um ein System zu ersetzen, aber viele dieser Möglichkeiten erübrigen sich, sobald man die Systemumgebung betrachtet. Das hat unterschiedliche wirtschaftliche wie technische oder unternehmenspolitische Gründe. Daher kann man sich auf wenige Szenarien konzentieren und diese Schritt für Schritt verfeinern, je genauer desto besser. Jedoch erfordert das viel Zeit, die meist nicht gegeben ist. Somit ist jede Entscheidung mit einem gewissen Risiko behaftet. Die Entscheidungsfindung mithilfe von Szenarien hilft jedoch dabei, dieses Risiko zu minimieren – und das ist das eigentliche Ziel dieser Arbeit. Entscheidungen kann man nicht mit richtig oder falsch klassifizieren, denn jede Entscheidung hat ihre Vor- und Nachteile.

Somit gilt es, nach und nach alle Möglichkeiten zu filtern, sich stets auf die besten zu konzentrieren, diese wieder zu verfeinern und am Ende die Gewissheit zu erlangen, den richtigen Weg zu beschreiten.

# 2 Grundlagen der Qualitätssicherung

Qualitätssicherung in Unternehmen ist eine breit gefächerte und komplexe Domäne. Deshalb werden in diesem Kapitel grundlegende Begriffe erörtert und jene Bereiche genauer erörtert, die für die Entscheidnungsfindung von grundlegender Bedeutung sind.

## 2.1 Begriff und Definition von CAQ

Der Begriff CAQ steht für Computer Aided Quality Assurance. Die Definition ist sehr weitläufig und umfasst eine ganze Menge an Prozessen und Tätigkeiten zur Qualitätssicherung. Unternehmen, die eine umfangreiche Qualitätssicherung aufgrund verschiedener Qualitätsnormen implementiert haben, sind auf Softwaresysteme angewiesen, die diese Aufgaben unterstützen. Es gibt zahlreiche Anbieter von CAQ Systemen, die unterschiedliche Bereiche abdecken. In der Praxis hat sich folgende Einteilung etabliert [3]:

- Prüfplanung
- Wareneingangsprüfung
- Warenausgangsprüfung
- Erstmusterprüfung
- FMEA
- Fertigungsprüfung SPC
- Reklamationsmanagement
- Prüfmittelverwaltung
- Prüfmittelfähigkeit

## 2.2 Module eines CAQ Systems

Ein CAQ System kann man in verschiedene Module einteilen, wobei die Grenzen zwischen den einzelnen Modulen verschwimmen können und die Module ineinander greifen, d. h. ein Modul ist auf Daten und Ergebnisse eines anderen angewiesen. Am Markt hat sich die Einteilung in folgende Module etabliert:

### 2.2.1 Prüfplanung

Die Prüfplanung ist die Planung der Qualitätsprüfungen der zu erzeugenden oder zu bearbeitenden Produkte, sowie die Überwachung der Betriebsmittel. Die Art und Weise, wie die Qualitätsprüfungen durchzuführen sind, werden mit der Prüfplanung festgelegt. Sie ist ein Teil der Qualitätsplanung im Unternehmen. Die Prüfplanung wird für die Wareneingangsprüfung, die Warenausgangsprüfung, die Erstmusterprüfung und vor allem für die Fertigungsprüfung durchgeführt. Das Qualitätswesen kann in verschiedene Arbeitsbereiche gegliedert werden, diese sind nach DIN 55250 Teil 11 als Qualitätsplanung, Qualitätsprüfung und Qualitätslenkung definiert.

- Qualitätswesen
    - Qualitätsplanung
    - Qualitätsprüfung
        - Prüfplanung
        - Prüfdurchführung
        - Prüfdatenverarbeitung
- Qualitätslenkung

Die Prüfplanung ist ein Aufgabenbereich der Qualitätsprüfung. Die Prüfplanung beginnt mit einer Wareneingangsprüfung, überwacht prozessbegleitend und kontrolliert am Ende mit der Warenausgangsprüfung das fertige Produkt.

Sie umfasst den gesamten Produktionsablauf vom Wareneingang bis zur Auslieferung des Produktes. Die Prüfplanung kann in verschiedene Aufgaben eingeteilt werden:

- Die Auswahl der Prüfmerkmale
- Einordnung der durchzuführenden Prüfungen in den Produktionsprozess
- Bestimmung der zu prüfenden Einheiten bzw. des mengen- oder zeitbezogenen Prüfintervalls
- Festlegung der Reihenfolge, in der die einzelnen Merkmale zu prüfen sind
- Ermittlung des anzuwendenden Prüfmittels
- Festlegung der Prüfmethode
- Ermittlung der Prüfzeitpunkte
- Ermittlung der notwendigen Prüfdokumentation
- Aufarbeitung der Hinweise für die Prüfdurchführung

Zusammen mit den Zeichnungen, Arbeitsplänen und den oben genannten Punkten kann die Prüfplanung durchgeführt werden. Hinzu kommen Normvorschriften, z. B. der ISO/TS 16494[2] für den Automotivbereich sowie Richtlinien und Auflagen der KundInnen. Auch die Daten aus vorangegangenen Produktionsaufträgen können mit einfließen. Eine wichtige Rolle spielt auch die Erfahrung des Qualitätsverantwortlichen, um mögliche Risiken in diesem Stadium zu berücksichtigen und mit in die Planung aufzunehmen.

Der Prüfplan legt schlussendlich alle Aktivitäten fest und definiert Art und Umfang der durchzuführenden Prüfungen, d. h. was wird wann, wo, wie, von wem und womit geprüft.

In Prüfanweisungen, die ebenfalls im Prüfplan festgelegt werden, wird festgehalten, wie eine Prüfung zu erfolgen hat und ob diese zusätzlich dokumentiert werden soll. Auch wird festgehalten, ob ein Bauteil vor der Prüfung einer speziellen Behandlung unterzogen werden muss, ob es z. B. gereinigt werden soll und wenn ja, womit.

Der Prüfplan ist unabhängig vom Auftrag und speziell auf ein Teil zugeschnitten. Später werden dann auftragsspezifische Daten hinzugefügt, z. B. welche Material charge wurde verwendet, auf welcher Maschine wurde ein Arbeitsschritt durchgeführt. Dadurch entsteht der Prüfauftrag.

In der Regel erfolgt die Prüfplanerstellung gemeinsam mit oder kurz nach der Arbeitsplanerstellung.

### 2.2.2 Wareneingangsprüfung

Am Anfang des Produktionsauftrags stehen die Materialien, die zur Produktion benötigt werden. Diese werden in der Wareneingangsprüfung bewertet und es wird festgelegt, ob die Ausgangsstoffe den gestellten qualitativen Anforderungen entsprechen. Diese Prüfung muss vor dem Beginn der Verarbeitung erfolgen, um eventuelle Mängel aufzudecken.

Aus der Wareneingangsprüfung werden verschiedene Informationen gewonnen. Diese werden herangezogen, um den Lieferanten zu bewerten. Es erfolgen Auswertungen bezüglich der Lieferqualität, Liefertreue, Termintreue etc. Daraus lässt sich nach VDA Band 2 die Lieferqualität ermitteln, das Ergebnis wird durch eine Qualitätszahl

ausgedrückt. Somit entstehen verschiedene Klassen von Lieferanten, die so unabhängig von ihrem Produkt bewertet und verglichen werden können.

### 2.2.3 Warenausgangsprüfung / Endprüfung

Dies ist die abschließende Prüfung der produzierten Ware vor der Auslieferung an den Kunden ähnlich der Wareneingangsprüfung. So werden die Lieferung an den Kunden dokumentiert und können später nachvollzogen werden. Die Warenausgangsprüfung ist meist sehr aufwendig und kostenintensiv. Fehler werden erst am Ende der Produktion und somit zu spät bemerkt. Daher wird die Prüfung immer mehr in die Produktion verlagert, damit Abweichungen früher erkannt und die Ursache an Ort und Stelle schneller behoben werden kann. So wird der Ausschuss sehr gering gehalten, was wiederum Kostenvorteile mit sich bringt.

Die Warenausgangsprüfung kennt verschiedene Ausprägungen: die 100% Kontrolle, die Kontrolle mittels Stichproben und die Skip-Lot Prüfung.

Die 100% Kontrolle ist sehr aufwändig und somit teuer. Sie kann nur bei kleinen Stückzahlen wirtschaftlich angewendet werden, bei Mittel- und Großserien kann sie nur noch mit Prüfautomaten wirtschaftlich erfolgen.

Bei der Stichproben-Kontrolle wird nur ein Teil der Bauteile überprüft. Im wesentlichen finden heute Stichprobensysteme nach DIN ISO 2859 und DIN 3951 Anwendung, aber natürlich auch werkseigene Vorschriften.

Eine Erweiterung der Prüfung mittels Stichproben ist die Skip-Lot Prüfung. Hier fließen Daten der SPC in die Prüfung ein. War der

Produktionsauftrag über einen bestimmten Zeitraum stabil und lag die Anzahl der Abweichungen unter einem bestimmten Wert so, kann der Stichprobenumfang vermindert werden oder die Endkontrolle ganz entfallen, was wiederum einen Kostenvorteil mit sich bringt.

### 2.2.4 Erstmusterprüfung

First sample report. Der Kunde bewertet seine LieferantInnen anhand von verschiedenen Möglichkeiten wie:

- Referenzen, z.B.: Die Zertifizierung nach ISO 9000
- Audits bei den LieferantInnen
- Freigabetests

Vor der Auslieferung des Teils werden den KundInnen Musterteile zur Verfügung gestellt. Diese werden anhand der vorher erstellten Spezifikation von den KundInnen überprüft und nach erfolgreicher Prüfung freigegeben. Dann werden die Musterteile mit einem Erstmusterprüfbericht ausgeliefert. Dieser Bericht besteht aus zwei Teilen. Der erste Teil enthält die allgemeinen Angaben zum Grund der Erstmusterprüfung, zur Art des Berichtergebnisses (z.B.: Mess-, Werkstoff- oder Funktionsbericht), zu den organisatorischen Daten sowie zur Entscheidung. Im zweiten Teil werden neben den Stammdaten folgende Informationen erfasst:

- Sollwerte, vorgegeben vom Besteller
- Istwerte, gemessen bei den LieferantInnen
- Istwerte, gemessen beim Besteller
- AT außerhalb der Toleranz

Liegen die Istwerte außerhalb der Toleranz, so wird diese Tatsache im Erstmusterprüfbericht gekennzeichnet und die notwendige

Entscheidung vermerkt. Folgende Entscheidungen können getroffen werden:

- Richtigstellen, in dem z. B. der/die LieferantIn das Maß ändert, um die Sollwerte zu erreichen.
- Begrenzt freigeben, die Freigabe erfolgt für eine begrenzte Stückzahl.
- Die Abweichung wird in die Zeichnung übernommen, das Teil wird nicht geändert, sondern die Sollwerte in der Zeichnung werden angepasst.
- Akzeptiere Abweichung, dabei wird die Abweichung einfach akzeptiert, ohne die Zeichnung anzupassen.

### 2.2.5 FMEA

Der Begriff Fehlermöglichkeits- und Einflussanalyse, abgekürzt FMEA [4], (engl. failure mode and effects analysis), bezeichnet eine analytische Methode, um potentielle Schwachstellen zu finden. Im Rahmen des Qualitätsmanagements wird die FMEA zur vorbeugenden Fehlervermeidung eingesetzt. Die FMEA wird insbesondere in der Design- bzw. Entwicklungsphase neuer Produkte oder Prozesse angewandt.

FMEA folgt dem Grundgedanken einer vorsorgenden Fehlerverhütung anstelle einer nachsorgenden Fehlererkennung und -korrektur (Fehlerbewältigung) durch frühzeitige Identifikation potentieller Fehlerursachen bereits in der Entwurfsphase. Damit werden ansonsten anfallende Kontroll- und Fehlerfolgekosten in der Produktionsphase oder gar im Feld (bei KundInnen) vermieden und die Kosten insgesamt gesenkt. Durch die systematische Vorgehensweise und die dabei gewonnenen Erkenntnisse wird zudem

die Wiederholung von Designmängeln bei neuen Produkten und Prozessen vermieden.

Die FMEA wurde 1980 als Ausfalleffektanalyse in die DIN 25448 aufgenommen und später ersetzt durch die DIN EN 60812[5]. Die Methodik der FMEA soll schon in einer frühen Phase der Produktentwicklung (Planung und Entwicklung) innerhalb des Produktlebenszyklus angewandt werden, da eine Kosten-/Nutzenoptimierung in der Entwicklungsphase am wirtschaftlichsten ist. Denn je später ein Fehler entdeckt wird, desto schwieriger und kostenintensiver wird seine Korrektur.

### 2.2.6 Fertigungsprüfung und SPC

Statistical Process Control (SPC) ist ein mächtiges Werkzeug, um die Qualitätssicherung in der Fertigung durchzuführen. Damit dieses Werkzeug angewandt werden kann, muss zuerst die Prozessfähigkeit gewährleistet werden und der Prozess beherrschbar sein. Das kann mithilfe analytischer Methoden erfolgen[6], welche in einem späteren Kapitel näher behandelt werden.

Zur Überwachung der laufenden Serienproduktion werden Qualitätsregelkarten (QRK) eingesetzt, durch die systematische und nicht im Vorfeld vermeidbare Einflüsse, wie beispielsweise Werkzeugverschleiß, erfasst und Korrekturmaßnahmen eingeleitet werden.

Die Anwendung der SPC vollzieht sich nach den Grundsätzen des Regelkreises[7]. Dabei ist der Prozess die Regelstrecke, das oder die zu fertigenden Merkmale sind die Regelgröße und der Bediener nimmt mit dem Hilfsmittel der QRK die Rolle des Reglers ein. Die Maschinengrößen sind als Stellgrößen zu betrachten.

In der QRK wird in regelmäßigen zeitlichen oder stückzahlbedingten Abständen mithilfe einer Stichprobe der zu überwachende Wert eingetragen. Es entsteht so ein zeitlicher Verlauf der Werte, die innerhalb vorgegebener Grenzen liegen müssen. Dabei werden drei Arten von Grenzen unterschieden:

- Toleranzgrenzen (OGW, UGW)
- Eingriffsgrenzen (OEG, UEG)
- Warngrenzen (OWG, UWG)

Hierbei gilt folgender Sachverhalt: OWG<OEG<OGW bzw. UGW<UEG<UWG.

Werden die Warngrenzen überschritten kann der Prozess verschärft beobachtet werden, die Eingriffsgrenzen stellen die Regelgrenzen dar. Werden diese überschritten, muss der Prozess nachgeregelt werden. Die Toleranzgrenzen stellen die Bauteiltoleranzen dar und sollten niemals überschritten werden, da sonst außerhalb der Spezifikation gefertigt wird. Zusätzlich können prozessbezogene Messwerte eingegeben werden. All diese Merkmale werden dokumentiert und müssen später bei Bedarf den KundInnen übergeben werden.

### 2.2.7 Reklamationsmanagement

Das Reklamationsmanagement behandelt die Beziehung zwischen KundInnen und Unternehmen. Dabei gibt es genaue Abläufe, die bei Reklamationen eingehalten werden müssen. Diese umfassen die Tätigkeiten Planen, Durchführen und Überwachen der Maßnahmen, die das Unternehmen in diesem Fall ergreift, um den Fehler bzw. die Reklamation zu behandeln.

Dadurch soll die Kundenzufriedenheit wieder hergestellt, der Fehler nach Möglichkeit beseitigt und in Zukunft sichergestellt werden, dass

Ähnliches vermieden wird. Die negativen Auswirkungen einer Rekamation, z. B. die Abwanderung von KundInnen, Imageschäden und Unzufriedenheit sollen minimiert werden. Wichtig dabei ist, dass jeder Schritt genau dokumentiert und der vorgegebene Ablauf eingehalten wird. Dafür gibt es genaue Vorgaben der Automobilindustrie.

Die eventuelle Verknüpfung mit anderen qualitätsrelevanten Daten ist dabei von großem Vorteil, da die Reklamation so in viel kürzerer Zeit abgearbeitet werden kann, was zu erheblich geringeren Qualitätskosten führt.

In der Industrie ist der Ablauf zur Reklamationsbearbeitung weitestgehend standardisiert. Sie basiert auf zwei Schwerpunkten, die in 8 Schritten durchlaufen werden:

- Faktenorientiertes System: Problemlösung, Entscheidungsfindung und Planung basieren auf echten Daten, die laufend gesammelt und überwacht werden.
- Abstellung der Grundursache: Anstatt nur die aufgetretenen Auswirkungen zu überdecken, wird die Grundursache des Problems behoben.

Im Folgenden werden die 8 Stufen des 8D Reports [8] aufgezählt, die der Reihe nach durchlaufen werden:

1. Gehe das Problem im Team an.
2. Beschreibe das Problem.
3. Veranlasse temporäre Maßnahmen zur Schadensbegrenzung und kontrolliere ihre Wirkung.
4. Ermittle die Grundursache(n) und beweise, dass es wirklich die Grundursache(n) ist/sind.

5. Lege Abstellmaßnahme(n) fest und beweise ihre Wirksamkeit: Suche nach allen möglichen Maßnahmen, durch die die Ursache(n) beseitigt und das Problem gelöst werden könnte.
6. Führe die Abstellmaßnahme(n) ein und kontrolliere ihre Wirkung: Erstelle einen Aktionsplan zur Einführung der gewählten Abstellmaßnahme(n) und lege gegebenenfalls flankierende Maßnahmen zur Absicherung fest.
7. Bestimme Maßnahmen, die ein Wiederauftreten des Problems verhindern.
8. Würdige Leistung und Erfolg des Teams.

Für interne Abweichungen, die das Produkt betreffen, kann es mitunter sehr aufwändig werden, einen kompletten 8D Report durchzuführen. Hier kann es sinnvoll sein, ein vereinfachtes Verfahren anzuwenden.

### 2.2.8 Prüf- und Messmittelverwaltung

Prüf- und Messmittel unterliegen, wie andere Werkzeuge auch, der Abnutzung. Somit ist es erforderlich, sie in regelmäßigen Abständen zu überprüfen und bei Bedarf neu zu kalibrieren[9]. Die Menge der verschiedenen Messmittel kann sehr groß sein, dementsprechend umfangreich ist auch der Verwaltungsaufwand. Moderne Systeme dokumentieren den Zeitpunkt und den Kalibriervorgang. Sie entscheiden anhand der entstehenden Prüfhistorie über das optimale Kalibrierintervall. So können die meist aufwändigen Kalibriervorgänge auf ein Minimum begrenzt werden. Das Messmittel kann häufiger eingesetzt werden. So wird auch sichergestellt, dass die Messmittel kalibriert werden, da das System von sich aus aktiv wird.

## 2.3 Qualitätswerkzeuge

Qualitätswerkzeuge lassen sich im Allgemeinen in 2 Kategorien einteilen, in die On- und Off-Line Qualitätswerkzeuge. SPC gehört zur ersten Kategorie und dient der Überwachung und Steuerung einer laufenden Serienproduktion. Die zweite Kategorie befasst sich mit der statistischen Auswertung von Versuchsreihen. Es ist ein sehr wichtiges Feld, das jedoch für das Thema der Arbeit von geringer Bedeutung ist und so nur der Vollständigkeit halber erwähnt wird.

In den 70er Jahren wurde die statistische Prozesslenkung (SPC = Statistical Process Control) neu belebt. Die seitdem geführten Diskussionen über Sinn und Unsinn können nicht darüber hinwegtäuschen, dass richtig und sinnvoll angewandte statistische Prozesskontrolle in vielen Bereichen der industriellen Fertigung unerlässlich ist. SPC wird heute nicht mehr in Frage gestellt und bildet vielmehr den Motor einer ständigen Verbesserung des Produktionsprozesses. Es ist ein System, das einen Fertigungsprozess in seinem Umfeld aufgrund der erhobenen Prozessdaten modellhaft beschreibt.

### Das Prinzip der ständigen Qualitätsverbesserung

Die Definition von Qualität[10] war ursprünglich durch die Einhaltung von Grenzwerten bestimmt. Ein Qualitätsmerkmal musste sich also innerhalb einer bestimmten Toleranz befinden. Diese Tatsache bietet aber kaum Anreize zur Verbesserung der Produktqualität, denn so lange der Messwert dieser Bedingung entspricht, ist niemand bestrebt,

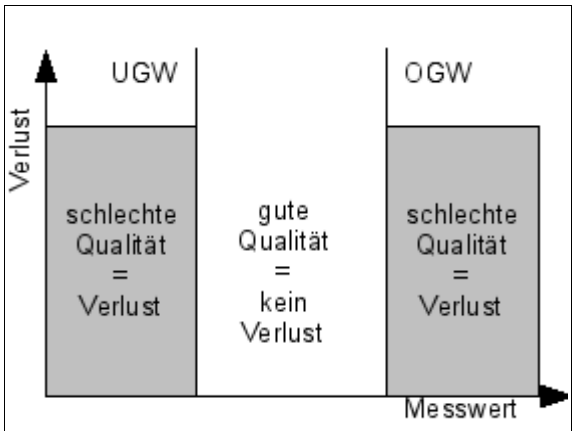


**Abbildung 2: Gut Schlecht Denken**

Verbesserungen am Produktionsprozess vorzunehmen. Das Verlangen der KundInnen richtet sich also immer mehr nach qualitativ hochwertigeren Erzeugnissen. Die Bestrebung muss sein, mit wirtschaftlich und technisch vertretbaren Anstrengungen eine möglichst geringe Streuung um den Zielwert eines Qualitätsmerkmals zu erreichen. Das bedeutet, der Produktionsprozess muss von allen Beteiligten ständig verbessert werden, um nahe an den Sollwert heranzukommen.

Eine Verbesserung kann aber nur erreicht werden, wenn man sich vom Gut-Schlecht-Gedanken verabschiedet, bei dem Teile, die innerhalb der Toleranz sind, als gute Teile und jene, die die Toleranz verletzten, als schlechte Teile eingestuft und dann im Allgemeinen verworfen werden.

Hierbei macht man jedoch keinen Unterschied zwischen einem Teil, dessen Ist-Wert nahe dem Sollwert liegt und einem Teil, das sich nahe der Toleranzgrenze befindet. Warum aber soll ein Teil, das knapp außerhalb der Toleranzgrenze liegt, gänzlich ungeeignet sein und verworfen werden und ein Teil, das knapp innerhalb der Toleranzgrenze liegt, genauso gut geeignet sein, wie jenes, dessen Sollwert nahe dem Istwert liegt? Im Bezug auf den Verwendungszweck ist dieses Auswahlverfahren also äußerst fragwürdig.

Vergleicht man ein Teil, dessen Toleranz knapp innerhalb des Grenzwertes liegt mit einem, das gerade außerhalb liegt, so kann man nicht verstehen, warum das eine uneingeschränkt geeignet ist und das andere hingegen vollkommen unbrauchbar sein soll. Beide Teile liegen mit ihren Ist-Werten nur geringfügig auseinander und bezogen auf ihren Verwendungszweck können sie niemals derartig große Unterschiede aufweisen, dass dieser kleine Abstand über vollkommenes Sein oder Nichtsein entscheidet.

Man benötigt also ein genaueres Bild zur Beschreibung der Qualität eines Wertes, als nur die einfache Gut- oder Schlecht-Einstufung. Dies erfolgt durch ein Modell, das die Abweichung durch den Zielwert in einer Verlustfunktion darstellt. Der Zielwert ist das Optimum der Kundenerwartung. Je weiter der Istwert vom Zielwert abweicht, umso höher ist der Verlust, also die Qualitätsbeeinträchtigung des Teils.

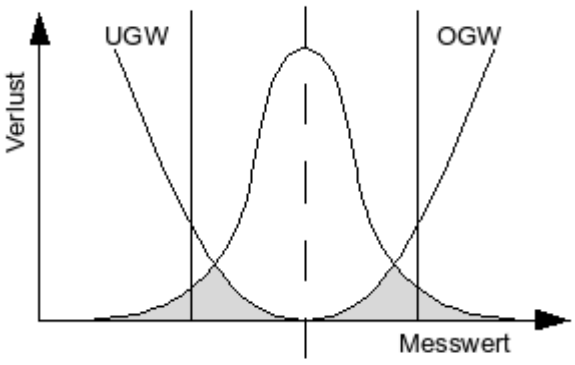


**Abbildung 3: Verlust in Abhängigkeit von Lage und Streuung**

Ziel der Fertigung muss es also sein, den Zielwert anzustreben, um so den Verlust möglichst gering zu halten. Die Verlustfunktion kann man als Parabel darstellen, deren Minimum beim Istwert liegt. Betrachtet man nun beispielsweise die Messwerte als Gaußsche Glockenfunktion, deren Mittelwert gleich dem Istwert ist, so erkennt man, indem man des Weiteren die Verlustfunktion und die Gaußsche Glocke überlagert, wie wichtig eine möglichst geringe Streuung der Messwerte ist. Je größer die Streuung, desto mehr Mess-werte findet man nahe der Toleranzgrenze. Diese Werte bedeuten aber den größten Verlust, die größte Qualitätsbeeinträchtigung. Je geringer also die Streuung, umso weniger Messwerte mit hohem Verlust sind vorhanden. Eine Reduktion der Streuung hilft also, den Verlust zu mindern.

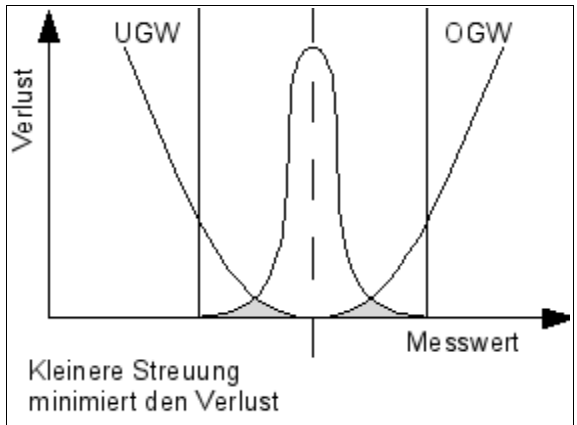


**Abbildung 4: Verbesserte Verlustabhängigkeit**

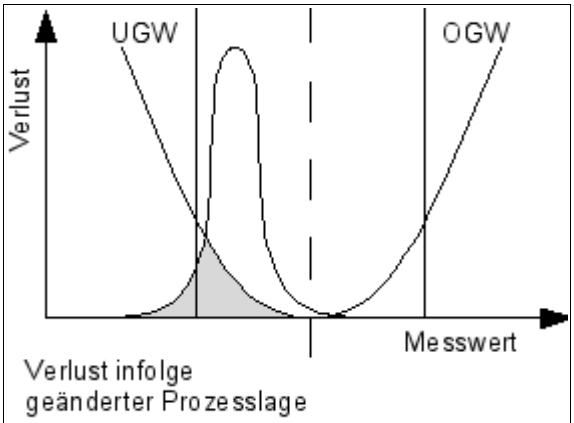


**Abbildung 5: Veränderte Prozesslage**

In weiterer Folge ist aber nicht nur die Streuung ausschlaggebend für die Höhe der Qualitätsbeeinträchtigung, sondern auch der Mittelwert. Je mehr der Mittelwert vom Zielwert abweicht, umso höher ist der Verlust, auch wenn die Streuung einen kleinen und somit verlustmindernden Wert erreicht hat.

Diese Modelle besitzen natürlich nicht den Anspruch auf Vollständigkeit bezogen auf einen Fertigungsprozess und erzeugen sicher in einigen Punkten Diskussionsstoff. Sie untermauern jedoch die Erkenntnis, dass die reine Orientierung an der Spezifikation eines Merkmals nicht dem Gedanken des Strebens nach einer ständigen Qualitätsverbesserung gerecht werden kann. Vielmehr müssen alle eingebundenen MitarbeiterInnen in einem Produktionsprozess ihr Tun und Handeln auf das Erreichen des Zielwerts ausrichten und gleichzeitig versuchen, die Streuung auf ein Minimum zu beschränken und ständig zu verkleinern.

## 2.4 Modellbeschreibung des Produktionsprozesses

Damit der Produktionsprozess modellhaft beschrieben werden kann, greift man auf statistische Verfahren zurück. Je genauer man den Sachverhalt mit diesen Verfahren beschreiben kann und je exakter die gemessenen Ergebnisse sind, umso wertvoller sind die

gesammelten Informationen. In der Fertigung werden dazu Prozessparameter und Merkmaleigenschaften von gefertigten Teilen möglichst exakt erfasst bzw. gemessen. Dazu ist eine ausreichend große Menge an gemessenen Teilen, also eine entsprechend große Stichprobe nötig, um dann das Verhalten der Maschine bzw. des Prozesses zu erfassen. Idealerweise würde man eine deterministische Beschreibung in Form einer Funktion f(xi) suchen, die den Zusammenhang zwischen dem Parameter xi und dem Funktionswert eindeutig beschreibt. Da dies aufgrund der komplexen Verhaltensweise und Strukturen nicht möglich ist, greift man auf ein statistisches Verfahren zurück. Man bildet also das reale System als mathematisches Modell möglichst exakt ab. Die Güte des Ergebnisses hängt im Wesentlichen von der Exaktheit dieses Models ab.

Es wurden nun verschiedene Prozessmodelle entwickelt, wobei hier die drei wichtigsten vorgestellt und die anderen im Allgemeinen von diesen abgeleitet werden. All diese Modelle bilden eine idealisierte Ausprägung und gehen in Wirklichkeit fließend ineinander über. Daher ist es schlussendlich dem Betrachter überlassen, für welches Modell er sich entscheidet.

Grundlegend ist unter einem Prozess eine fortlaufende Serienproduktion zu verstehen. Dieser werden in bestimmten Intervallen Stichproben entnommen, die dann wiederum Aufschluss darüber geben, wie sich der Prozess über die Zeitachse verhält. Im Folgenden werden die 3 wichtigsten Modelle vorgestellt:

### 2.4.1 Prozessmodell A

Dieses Modell ist durch folgende Merkmale gekennzeichnet:

- Zu deinem bestimmten Zeitpunkt (t) entspricht die Verteilung der Merkmalwerte einer Normalverteilung mit der Standardabweichung σ(t) und dem Mittelwert μ(t).
- Die Prozessstandardabweichung σ(t) ist konstant.
- Der Prozess Mittelwert μ(t) ist konstant.

Für den Prozess gilt, die Gesamtstandardabweichung σges = ^σ . Dieses Modell kommt in der Praxis kaum vor und stellt einen Idealfall dar (klassisches Shewhart-Modell).

Es sei noch angemerkt, dass bei einseitig physikalisch begrenzten Merkmalen im Allgemeinen keine Normalverteilung vorliegt. sondern meist eine Logarithmische Verteilung.

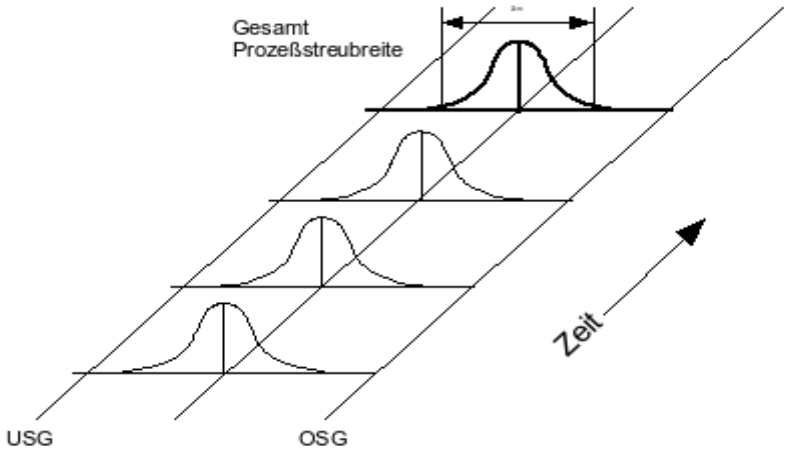


**Abbildung 6: Prozessmodell A**

### 2.4.2 Prozessmodell B

- die momentane Verteilung zu einem Zeitpunkt (t) entspricht einer Normalverteilung mit der Standardabweichung σ(t) und dem Mittelwert μ(t).
- Standardabweichung σ(t) ist konstant
- Die Prozessmittelwerte μ(t) streuen sich normal verteilt um einen langfristig konstanten Mittelwert.

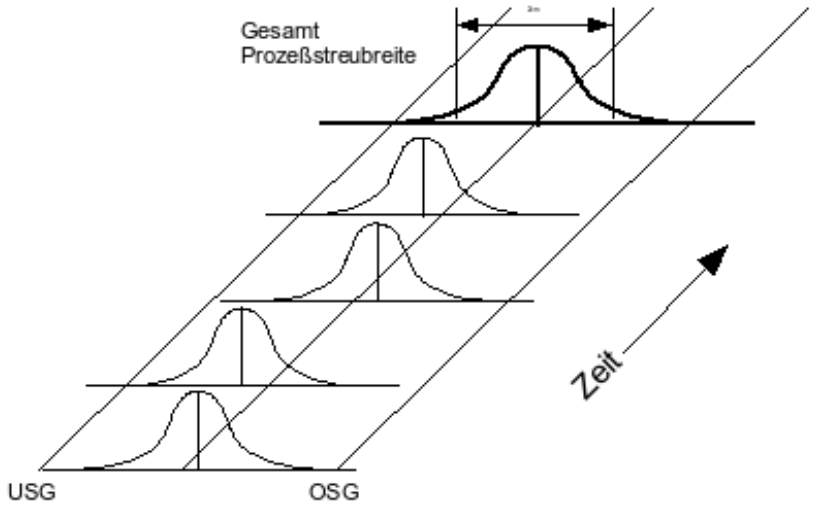


**Abbildung 7: Prozessmodell B**

Derartige Schwankungen kommen am häufigsten vor, man erkennt hier deutlich dass sich die Streuung der einzelnen Stichproben nicht verändert wohl aber deren Lage. Somit kann der Prozess kontrolliert werden indem man vor allem den Mittelwert berücksichtigt.

### 2.4.3 Prozessmodell C

- Die momentane Verteilung zu einem Zeitpunkt (t) entspricht einer Normalverteilung mit der Standardabweichung σ(t) und dem Mittelwert μ(t).
- Die Standardabweichung σ(t) ist konstant.
- Die Prozessmittelwerte μ(t) ändertn sich nach bekannten Gesetzmäßigkeiten.

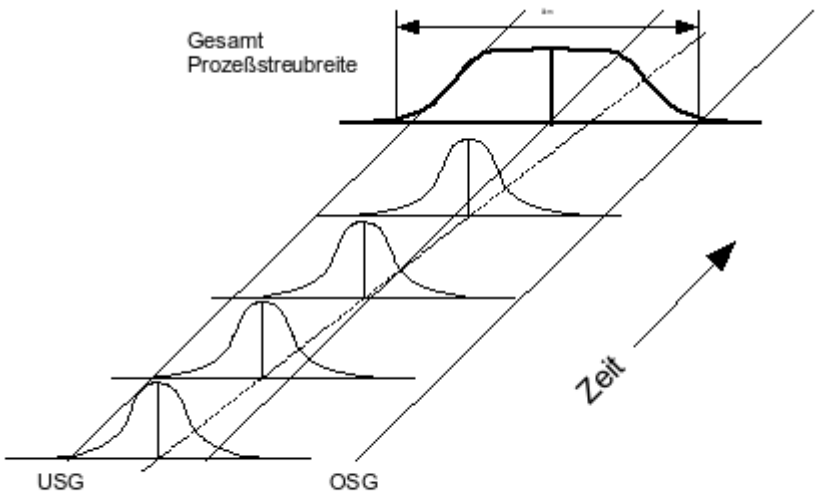


**Abbildung 8: Prozessmodell C**

Dieses Modell kommt z.B. bei Werkzeugverschleiss vor, der Mittelwert verändert sich kontinuierlich in eine bestimmte Richtung durch die Abnutzung eines Werkzeugs. Die gesamte Prozeßstreubreite ist zwar entsprechend hoch die Streuung der einzelnen Stichproben ist aber klein und somit kann ein derartiger Prozess gut kontrolliert werden wenn man die Verschiebung der Mittelwerte ausreichend beobachtet.

## 2.5 Klassifizierung von Qualitätsmerkmalen

Produkte in ihren verschieden Ausprägungen als Teile, Elemente, etc. werden durch bestimmte Eigenschaften charakterisiert und beurteilt. Diese Produkteigenschaften kann man grundlegend in zwei Gruppen einteilen.

- quantitative Merkmale
- qualitative Merkmale

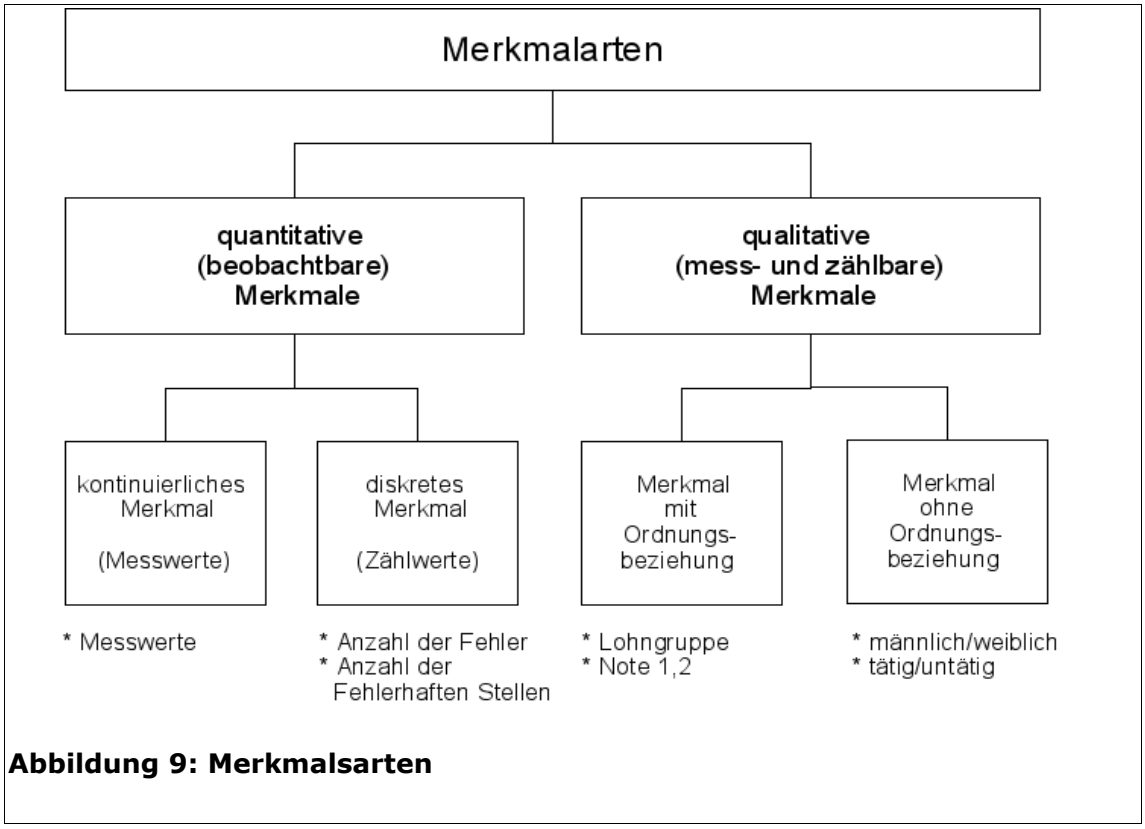


**Abbildung 9: Merkmalsarten**

### Qualitative Merkmale

Qualitative Merkmale sind mess- oder zählbar. Sie werden aufgrund dieser Eigenschaft nochmals in kontinuierliche und diskrete Merkmale unterteilt.

Kontinuierliche, also veränderliche Merkmale liefern bei der Beobachtung Messwerte. Werden diese Messwerte nun auf einer Skala dargestellt, kann jeder beliebige Punkt der Skala mit einem Messwert belegt werden. Beispiele für kontinuierliche Merkmale wären der Schwingerdurchmesser eines Lagers, das Alter einer Person oder der Bremsweg eines Fahrzeugs.

Die Beobachtungswerte von diskret veränderbaren Messwerten sind ganzzahlig. Trägt man die Messwerte auf einer Skala ein, so kann niemals ein Beobachtungswert zwischen den aufeinander folgenden Zahlen der Skala liegen.

Beispiele sind die Anzahl der Bruchstellen pro Einheit, die Anzahl der Fehlstücke einer Stichprobe oder die Anzahl der gefertigten Teile pro Stunde.

Ein Messwert kann sowohl attributiv als auch kontinuierlich erfasst werden. Wird die genaue Zahl erfasst, spricht man von einem kontinuierlichem Merkmal. Wird die Tatsache erfasst, ob sich der Wert innerhalb der Toleranzgrenze befindet, spricht man von einem attributivem Merkmal. In diesem Zusammenhang findet man auch den Begriff der Gut-Schlecht-Prüfung.

**Quantitative Merkmale**

Quantitative Merkmale sind beobachtbare Merkmale, die eine oder keine Ordnungsbeziehung besitzen. Aufgrund dieser Eigenschaft werden sie abermals unterteilt in Ordinal- und Nominalmerkmale.

Ordinalmerkmale liegen vor, wenn die Beobachtung eines Merkmalträgers ein Urteil liefert. Dieses wird mittels einer Ordinalskala erfasst. Die Abstände sind dabei beliebig wählbar. Als typisches Beispiel kann eine Gut-Schlecht, Prüfung angefügt werden, die mit Hilfe einer Lehre durchgeführt wird. Das Ergebnis wird dabei in gut oder schlecht eingeteilt. Ein weiteres Beispiel mit Ordungsbeziehung ist das Notensystem 1,2 etc.

Nominalmerkmale liegen vor, wenn für die Beschreibung eines Sachverhalts für den Wert des Merkmals keine Ordnungsbeziehung vorliegt. Das Ergebnis wird einer Nominalskala zugeordnet, die Einteilung unterliegt dabei einer Rangordnung. Beispiele sind die Einteilung des Geschlechts in weiblich/männlich, oder die Kostenart in Material-, Maschinen- Verwaltungs- und Sonderkosten.

## 2.6 Darstellung von Messergebnissen

Die einzelnen Messwerte für sich bilden zwar den Grundstein der Auswertung, sie werden erfasst und müssen innerhalb kürzester Zeit interpretiert werden, doch das ist durch eine reine Zahlendarstellung schwer möglich. Daher wurden verschiedene Diagramme entwickelt, um diese Zahlen in eine graphische, leicht und schnell interpretierbare Form zu bringen. Im Folgenden werden die wichtigsten Darstellungsmöglichkeiten beschrieben.

### 2.6.1 Darstellung von Einzelwerten

Um die gemessenen Werte sinnvoll beurteilen zu können, ist es notwendig, die Messergebnisse in einer übersichtlichen grafischen Form darzustellen[11]. Die einfachste und gleichzeitig auch aussagekräftigste Darstellung ist der Verlauf der Einzelwerte. Dabei werden auf der Ordinate die jeweiligen Einzelwerte eingetragen. Auf der Abszisse erfolgt die Nummerierung der Werte mit Datum und/oder Zeit, Chargennummer oder Nummerierung der Messdaten. Welche Einteilung gewählt wird, hängt dabei von der Art der Anwendung ab. Zusätzlich werden für die Messergebnisse relevante

Ereignisse notiert, d. h. Ereignisse, die die Messergebnisse beeinflussen können wie z. B. Werkzeugwechsel, Materialwechsel, etc.

## 2.6.2 Wertestrahl

In diesem Diagramm werden alle Einzelwerte unabhängig vom zeitlichen Verlauf betrachtet. Es werden die absoluten Häufigkeiten

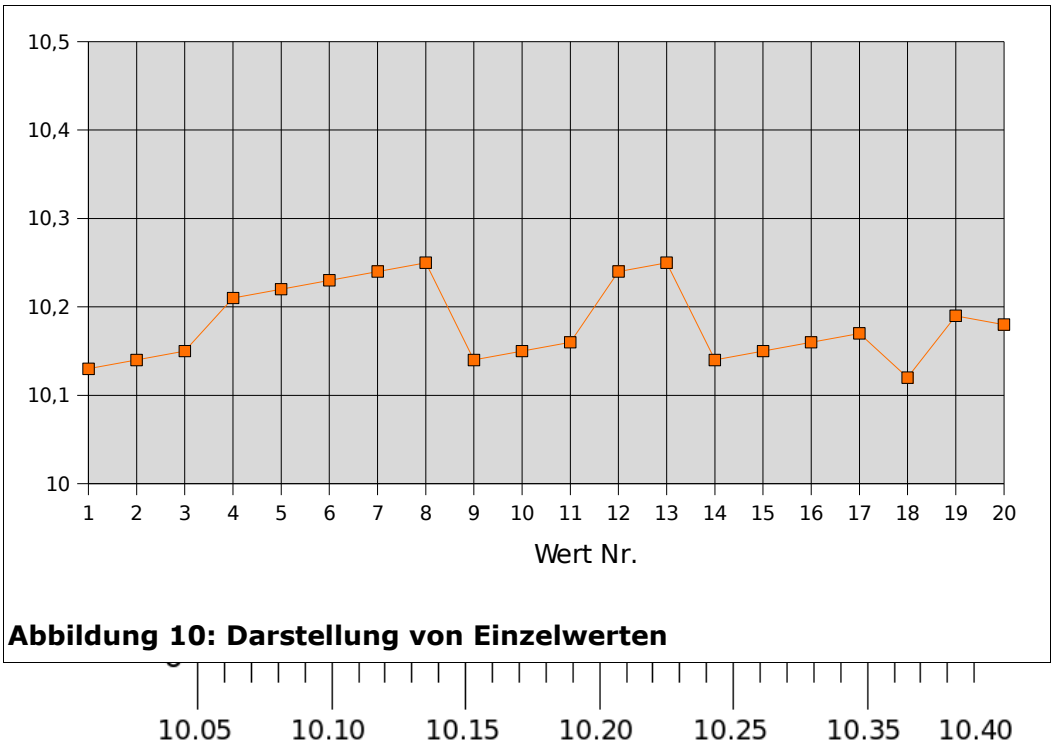


**Abbildung 10: Darstellung von Einzelwerten**




**Abbildung 11: Wertestrahl**

eines Wertes auf der Abszisse aufgetragen, die Einteilung der Ordinate erfolgt über die Auflösung des Messmittels. Wird z. B. ein Messschieber mit einer Auflösung von 0,01 mm verwendet, erfolgt die Skalierung in 0,01 mm Schritten. So geht keine Information durch Klassifizierung verloren, im Gegensatz zu einem

Häufigkeitsdiagramm, in dem eine beliebige, aber sehr wohl sinnvolle Klassifizierung der Messwerte vorgenommen werden muss.

Zusätzlich werden noch die Spezifikationsgrenzen eingetragen in den +/-3s Bereich. Dieser ist nur bei einer normal verteilten Messreihe sinnvoll, ansonsten wird der obere und untere Prozentpunkt eingetragen.

### 2.6.3 Histogramm

Ein Histogramm entsteht durch die Klassierung der x-Achse im Wertestrahl. Auf der Abszisse werden die absoluten oder relativen Häufigkeiten der Messwerte eingetragen, bezogen auf die Gesamtzahl der Stichproben. Die Ordinate zeigt die Klasseneinteilung. Die Anzahl der Werte, die in eine Klasse fallen, wird dabei als Klassenbesetzungszahl bezeichnet. Im Gegensatz zum Wertestrahl gehen bei dieser Methode geringfügig Informationen verloren, dafür verbessert sich die Übersichtlichkeit und die Möglichkeit, Vergleiche zu verschiedenen Messreihen zu ziehen. Es gibt verschiedene Verfahren, die Klasseneinteilung zu bestimmen. Hier liegt auch die größte Herausforderung, denn nur mithilfe einer einheitlichen Vorgehensweise können Auswertungen später miteinander verglichen werden. Die Einteilung kann z. B. nach DIN- Norm 55302 erfolgen.

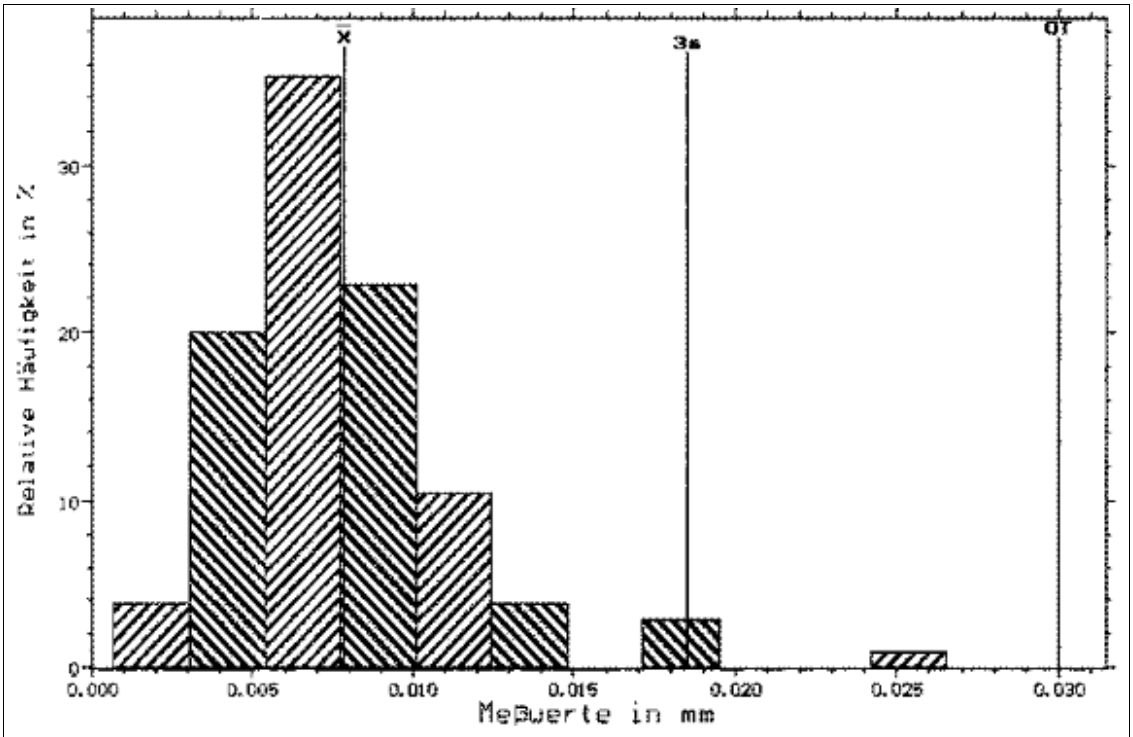


**Abbildung 12: Histogramm**

Auch in der CNOMO-Norm [12] gibt es Vorschriften, wie eine Einteilung vorgenommen werden muss. Darüber hinaus gibt es eine Reihe weiterer Möglichkeiten, wobei stets wichtig ist, dass die Einteilung nach einem bestimmten Verfahren erfolgt, nachvollziehbar ist und, dass ein Vergleich der Messreihen möglich ist.

### 2.6.4 Darstellung von statistischen Kennwerten

In einem laufenden Produktionsprozess werden Stichproben entnommen. Dabei kann der Mittelwert und die Varianz der einzelnen Stichproben berechnet werden. Zur Beurteilung der Grundgesamtheit über alle Stichproben hinweg, kann dann die Frage gestellt werden, ob:

- die Mittelwerte und Varianzen zufällig,
- die Mittelwerte selbst normal,
- die Varianzen gleich und
- die Varianzen $X^2$

verteilt sind.

Analog zu den Einzelwerten kann auch der Verlauf der Mittelwerte und der Varianzen im Laufe der Zeit dargestellt werden. Diese können zusätzlich mit sogenannten "gleitenden Mittelwerten" überlagert werden. Dabei werden einige Werte links und rechts vom aktuellen Wert zusammen mit diesem zur Bildung eines gemeinsamen Mittelwerts herangezogen. Durch dieses Verfahren erhält man eine Glättung des Verlaufs, d. h. zufällige Ereignisse und Ausreißer sind nicht mehr erkennbar.

Eine der wichtigsten Darstellungen von Stichprobenwerten ist die Qualitätsregelkarte. Ihr ist das folgende Kapitel gewidmet.

## 2.7 Qualitätsregelkartentechnik

In der Qualitätsregelkartentechnik[13] werden die erfassten Messwerte in eine grafische Form gebracht, die es ermöglicht, die Stichproben schnell und einfach zu interpretieren. Im Folgenden wird der grundlegende Aufbau der Regelkarten im Allgemeinen und die wichtigsten Regelkarten im Speziellen vorgestellt.

### 2.7.1 Aufbau und Interpretation einer Regelkarte

Die Qualitätsregelkarte visualisiert das Prozessverhalten bezüglich seiner Lage und Streuung. Dazu werden die Kennwerte herangezogen und über eine Zeitspanne dargestellt. Kennwerte sind hierbei z. B.: Anzahl der fehlerhaften Einheiten, Anzahl der Fehler pro Einheit, Urwerte, Mittelwerte, Mediane (Zentralwerte), Standardabweichungen und Spannweiten zur Lage- und Streuungsbeurteilung. Jeder Kennwert wird mit Grenzlinien, sog. Eingriffsgrenzen versehen. Die Eingriffsgrenzen sind weit enger gesetzt als die Toleranzgrenzen, sodass eine Verletzung einer Eingriffsgrenze einen Hinweis darauf liefert, dass der Prozess aus dem Ruder läuft und eventuell Maßnahmen ergriffen werden müssen, um den Prozess wieder zu stabilisieren.

Folgende Grafik zeigt den Aufbau einer Regelkarte, die dann im Folgenden im Detail beschrieben wird:

Auf der horizontalen Achse (Abszisse) wird

- die Nummer der Stichprobe,
- der Zeitpunkt (Datum/Urzeit) der Stichprobenentnahme oder
- die Chargennummer bzw. eine sonstige Kennzeichnung

verwendet.

Auf der vertikalen Achse (Ordinate) werden die Merkmalsausprägungen eingetragen. Dabei muss man zwischen kontinuierlichen und diskreten Merkmalen unterscheiden.

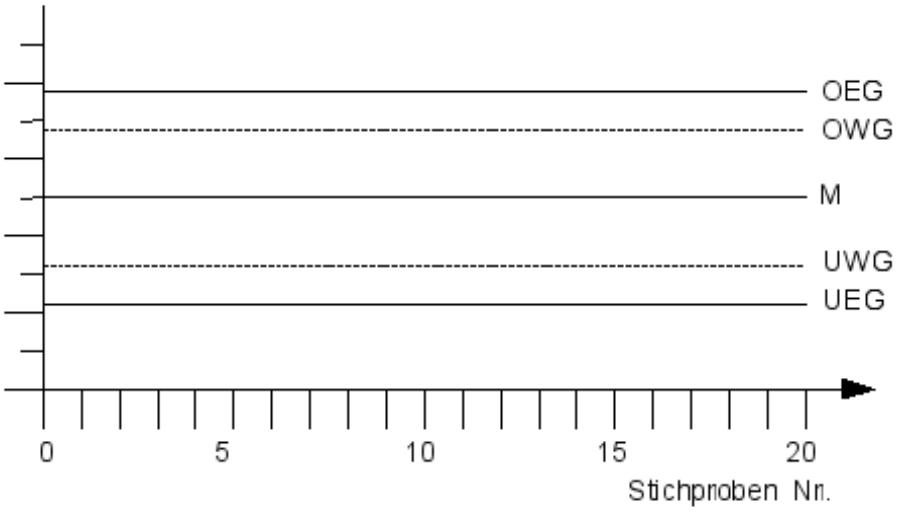


**Abbildung 13: Leere Regelkarte**

Bei kontinuierlich veränderlichen Merkmalen bestimmt der jeweilige Kennwert die Skalierung der Ordinate. Das kann eine Skala für die Urwerte oder eine für die statistischen Kennwerte der Stichprobe sein, z. B.: Median, Mittelwert, Standardabweichung oder Spannweite (Range).

Bei diskreten Merkmalen wird entweder die Anzahl der fehlerhaften Teile oder die Anzahl der Fehler pro Einheit eingetragen.

Weiters können je nach Bedarf folgende Kennlinien in die Regelkarte eingezeichnet werden:

- OEG – obere Eingriffsgrenze
- OWG – obere Warngrenze
- M – Mittellinie
- UWG – untere Warngrenze
- UEG – untere Eingriffsgrenze

Die oben dargestellte Regelkarte ist leer, d. h. es wurden noch keine Messungen durchgeführt bzw. Stichproben entnommen. In vorher

genau festgelegten Abständen (z. B. Zeit, Stückzahl) werden Stichproben mit dem Umfang n entnommen. Für jedes Merkmal wird eine eigene Regelkarte erstellt. Bei einem diskreten Merkmal wird die Anzahl der fehlerhaften Stück bzw. die Anzahl der Fehler pro Stück festgestellt und eingetragen. Das Stichprobenumfang kann dabei, im Gegensatz zu kontinuierlichen Merkmalen, bei denen der Stichprobenumfang immer konstant bleiben muss, variieren. Das Stichprobenumfang muss mindestens n = 3 betragen, die Praxis zeigt, dass ein Probenumfang von n = 5 sinnvoll ist. Von einer Stichprobe werden ein oder mehrere Merkmale geprüft. Aus den n Urwerten können des weiteren die verschiedenen statistische Kennzahlen errechnet werden. Je nach Qualitätsregelkartentyp werden die Urwerte selbst oder die errechneten statistischen Kennzahlen eingetragen. Folgende Abbildung zeigt die Mittelwerte eines kontinuierlichen Merkmals in einer sogenannten Mittelwertkarte (Xquer-Karte).

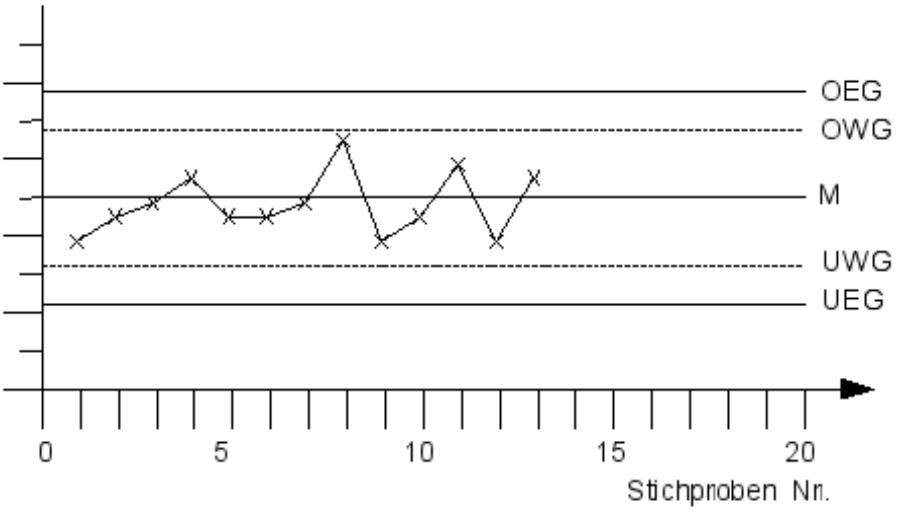


**Abbildung 14: Beispiel Regelkarte**

Nach jeder Messung erfolgt eine Beurteilung der Regelkarte. Dabei können drei Fälle unterschieden werden:

1. Ur- oder Kennwerte sind innerhalb der Warngrenzen, die Fertigung kann wie bisher weiterlaufen.
2. Ur- oder Kennwerte sind außerhalb der Warngrenzen aber innerhalb der Eingriffsgrenzen. Daraus folgt, dass die Fertigung wie bisher weiter läuft aber häufiger geprüft werden sollte, bzw. eventuell sofort eine weitere Stichprobe entnommen wird.
3. Ur- oder Kennwerte sind außerhalb der Eingriffsgrenzen, in diesem Fall muss die Fertigung neu eingestellt, bzw. sogar die Teile seit der letzten Prüfung aussortiert werden.

Hinzu kommen weitere Kriterien von geringerer Bedeutung, die jedoch auch beachtet werden können und deshalb hier kurz erwähnt werden.

**Run**

Eine Folge von 7 Werten liegt unterhalb oder oberhalb der Mittellinie, dies gilt für eine Xquer sowie s-Karte. Bei diesen Karten liegen erwartungsgemäß 50 % der Werte, sofern der Prozess ungestört und normal verteilt ist, unterhalb bzw oberhalb der Mittellinie. Die Wahrscheinlichkeit, dass es sich um einen Fehlalarm handelt, wenn 7 Werte unter- bzw. oberhalb der Mittellinie liegen, ist kleiner als 1 % (Fehler 1. Art).

**Trend**

Eine Folge von Werten steigt kontinuierlich an oder ab. Geht man von einer zufälligen Anordnung der Kennwerte aus, so ergeben sich für k Stichproben k(!) Möglichkeiten einer Anordnung. Geht man weiters davon aus, dass es keinen Kennwert gibt, der exakt gleich ist, gibt es genau eine Möglichkeit, dass eine Folge ab- bzw. aufsteigend ist. Die Wahrscheinlichkeit beträgt somit für drei Werte 1/6, für 5 Werte 1/120 = 0.833 % – also weniger als 1% – und wäre somit eine sinnvolle Trendanzeige. Da aber nur extreme Trendprozesse

signalisiert werden sollen, werden 8 Werte und somit 7 Intervalle verwendet.

**Middle Third**

Von 25 Stichproben müssen mindestens 40 %, aber nicht mehr als 90 % im mittleren Drittel liegen. Voraussetzung dafür ist, dass die Kennwerte einer normalverteilten Grundgesamtheit ohne Mittelwertschwankung entstammen, dass die Grenzen der Qualitätsregelkarte aus den -/+ 3 Sigma Grenzen der Kennwertverteilung bestimmt sind und, dass immer die gleiche Anzahl an Stichproben in der Regel k = 25 herangezogen wird.

## 2.7.2 Beispiele Qualitätsregelkarten

Bei qualitativen Merkmalarten kommen Fehlersammelkarten zum Einsatz. Abhängig von der Merkmalsart werden die Regelkarten bei quantitativen Merkmalen hingegen in jene für kontinuierliche und jene für diskrete Merkmaltypen eingeteilt. Bei kontinuierlichen Merkmalen wird eine Kombination aus Lage- und Streuungskarten verwendet. Bei den Lagekarten werden die Spezifikationsgrenzen heute nicht mehr berücksichtigt, denn nur so wird man dem Gedanken der ständigen Qualitätsverbesserung gerecht, immer möglichst nahe an den Ist-Wert heranzukommen. Die Regelkarten bestehen somit aus einer Lage-Spur und einer Streuungs-Spur. Typische Kombination hier sind die x/s Karte, Xquer/s Karte oder die xWelle/R Karte. Diese drei, sowie die Urwert- und die Fehlersammelkarte werden im Folgenden genauer vorgestellt.

**Fehlersammelkarten**

Bei Prozessen wird meist mehr als nur ein qualitatives Merkmal verwendet bzw. geregelt. Es liegt nahe, die Daten der einzelnen

Merkmale in einer gemeinsamen Karte – der Fehlersammelkarte – darzustellen.

Im Folgenden ist ein Beispiel einer solchen Karte dargestellt. Auf der linken Seite ist die Liste der Merkmale zu erkennen. Bei variablen Merkmalen ist in einer Spalte daneben der Messwert, bei attributiven Merkmalen die Anzahl der fehlerhaften Teile zu sehen, sofern es solche gegeben hat. Ansonsten symbolisiert ein Haken, dass die Messung in Ordnung war.

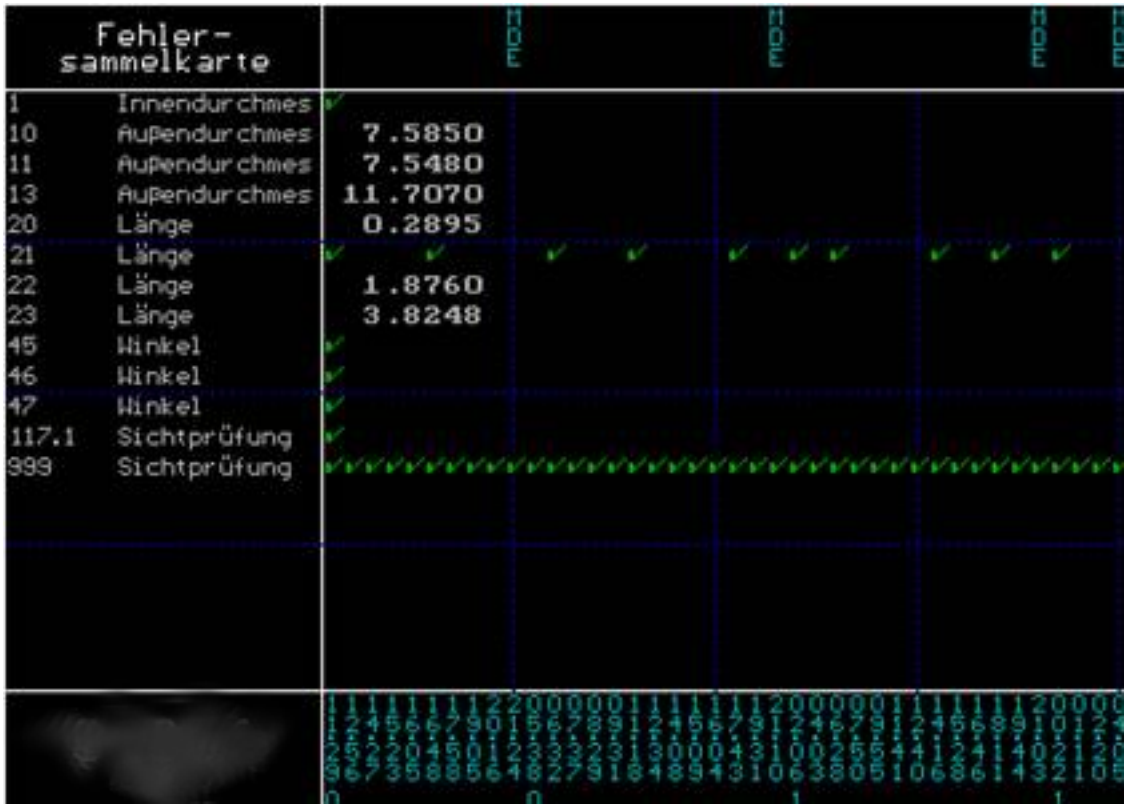


**Abbildung 15: Fehlersammelkarte**

## Urwertkarte

Die Urwertkarte ist wohl die einfachste Karte, da in ihr die einzelnen Urwerte der Stichprobe eingetragen werden. Sie werden nicht wie bei den anderen Karten zu einem Prüfwert (z. B. Mittelwert) weiterverrechnet. Urwertkarten werden nur beschänkt eingesetzt, da sie nicht so empfindlich auf Schwankungen reagieren wie andere

Karten. Trotzdem ist es oft sinnvoll, solche Karten einzusetzen, speziell in der Umformtechnik.

Im oberen Beispiel werden die Messwerte in 5er-Gruppen angezeigt. Jede horizontale Linie zeigt den Messwert an, jede vertikale Linie zeigt an, um den wievielten Messwert aus der 5er-Gruppe es sich handelt. In anderen Worten, jedes vertikale „Minuszeichen" wird in 5 Teile zerlegt. Links findet man den ersten Messwert, in der Mitte den dritten und rechts den fünften.

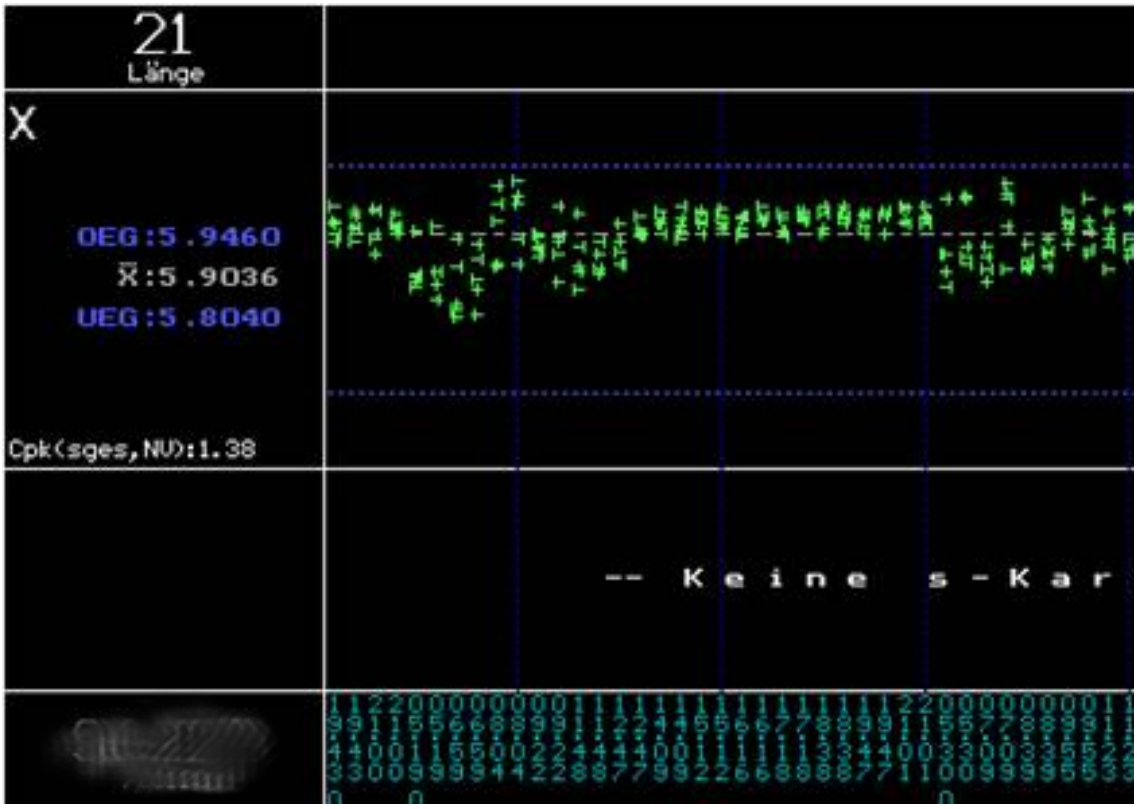


**Abbildung 16: Urwertkarte**

### /s Karte

Bei der s Karte werden beide Werte der Stichprobe, der Mittelwert und die Streuung in einem Diagramm dargestellt. Das bringt folgenden Vorteil, dass der Mittelwert der Stichprobe in Ordnung sein kann, die Werte aber sehr unterschiedlich waren, also eine sehr hohe Streuung aufweisen. Somit ist diese Stichprobe nicht mehr in Ordnung, da laut Wahrscheinlichkeitsrechnung in der gesamten Menge der Teile, sich einige befinden die den Qualitätskriterien nicht mehr stand halten.

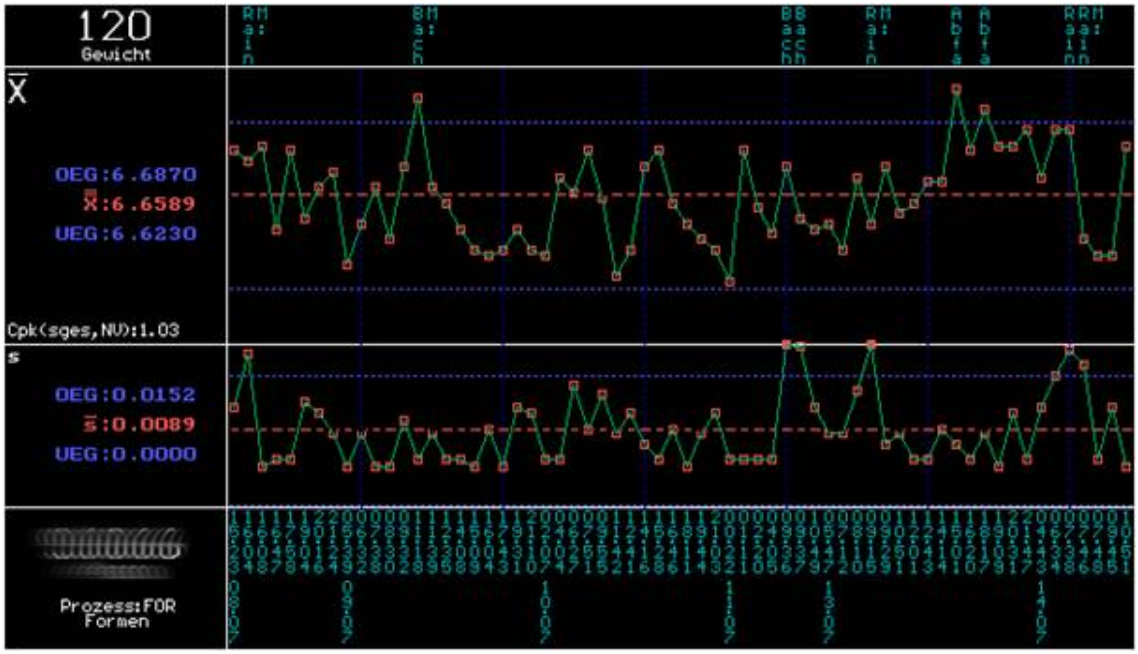


**Abbildung 17: X (quer) / S - Karte**

Auf der Abbildung erkennt man deutlich einige Fälle in denen die Eingriffsgrenzen verletzt beim Mittelwert verletzt wurden, obwohl die Streuung relativ klein war, umgekehrt ist der Mittelwert nahe dem Nominalwert, aber die Streuung ist so groß, dass wahrscheinlich einige der Teile außerhalb der Toleranzgrenzen liegen.

## /R Karte

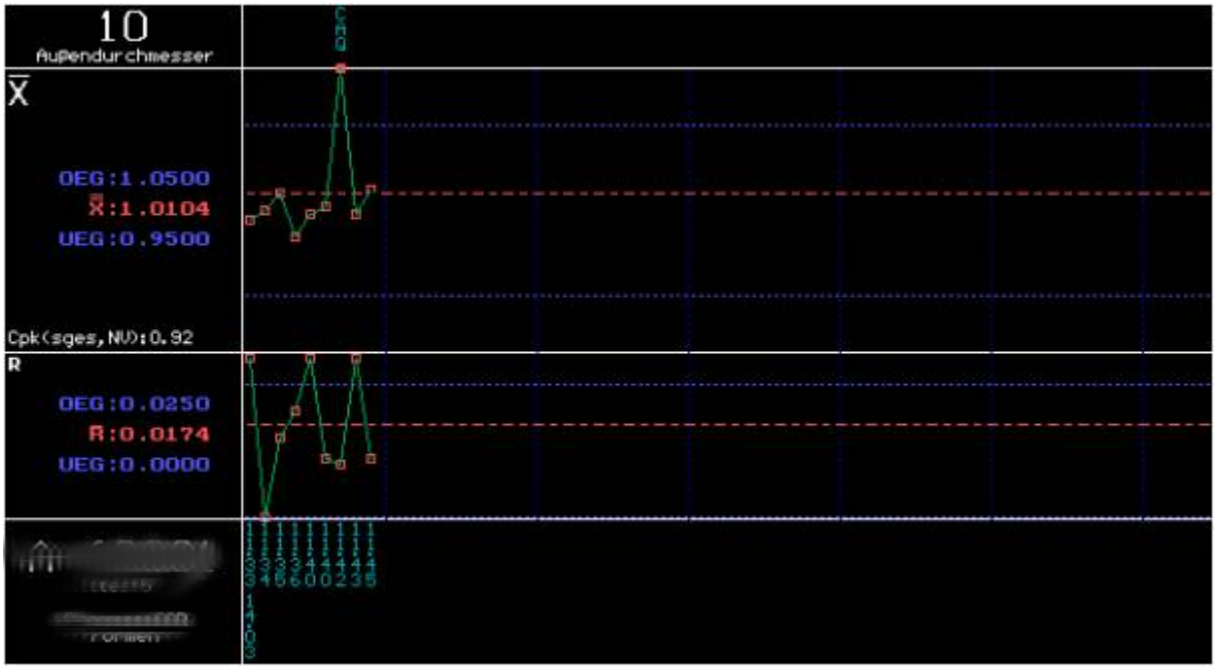


**Abbildung 18: X (quer)/R - Karte**

Die R Karte, stellt den Mittelwert und die Range in Relation dar, Range aus dem englischen bedeutet hier der Abstand vom grössten und kleinsten Messwert der Stichprobe. Die Karte ist wie die /s Karte aufgebaut, gibt aber anstelle der Streuung eben die Range an.

# 3 Analyse und Klassifizierung von Legacy Systems

Die Entscheidung, wie ein Legacy System verbessert werden soll, kann nicht innerhalb weniger Minuten getroffen werden. Vielmehr benötigt man dazu eine definierte Vorgehensweise, um die richtige Entscheidung zu treffen, die später auch begründet werden kann und nicht als eine reine Bauchentscheidung abgetan werden wird. Die richtige Entscheidung wird schlussendlich von vielen Faktoren abhängen. Es werden nicht nur direkte und indirekte Kosten berücksichtigt, auch potentielle Risiken müssen analysiert werden. Außerdem muss der Mehrwert, den die Software für ein Unternehmen darstellt, ermittelt werden. Darüber hinaus müssen auch politische Interessen der verschiedenen Stakeholder gewahrt werden. Man benötigt also eine festgelegte Vorgehensweise, um dann eine adäquate Entscheidungsgrundlage zu erhalten. Im ersten Kapitel wird der Ausdruck Legacy System genauer analysiert und definiert. Im zweiten wird eine Möglichkeit aufgezeigt, derartige Systeme zu klassifizieren und in einem dritten Kapitel werden dann Möglichkeiten für das Management dieser Systeme behandelt.

## 3.1 Software Wartung

So wie sich bei jedem technischem System irgendwann Verschleißerscheinungen bemerkbar machen, so kann das auch auf Software übertragen werden. Bits und Bytes altern zwar nicht, sie rosten nicht und wechseln dadurch ihren ursprünglichen Zustand – trotzdem spricht man auch im Softwarelebenszyklus von Wartung.

Wartung in dem Sinn, dass die Anforderungen, die an die Software bzw. an das System gestellt werden, Änderungen unterliegen und die Abläufe nicht mehr wie geschmiert mit dem System abgewickelt werden können. Deshalb muss das System eben entsprechend angepasst, also gewartet werden. Das folgene Kapitel geht auf diese Thematik genauer ein.

### 3.1.1 Abstraktionsgrade und -ebenen

Die ANSI Definition von Software Wartung ist: „Die Modifikation von Software nach dem Zeitpunkt der Auslieferung mit dem Zweck Fehler zu beheben, die Performance oder andere Eigenschaften zu verbessern oder um das Programm an die veränderte Umgebung zu adaptieren." [14] Reverse Engineering ist dabei ein Teil der Wartung und hilft, Aufbau und Funktionsweise des Systems zu verstehen, damit die entsprechenden Veränderungen vorgenommen werden können.

Um die Begriffe wie Reverse Engineering erklären zu können, muss grundsätzlich der Lebenszyklus der Software definiert werden. Chikofsky und Cross klassifizieren in „Reverse Engineering und Design Recovery" [15] die verschiedenen Begriffe, dabei verwenden sie als Basis das klassische Wasserfall Modell. Dieses kann bezüglich der Begriffsbildung auf verschiedene andere Modelle übertragen werden, vorausgesetzt es handelt sich um gerichtete Graphen, enthält also keine Zyklen. Prinzipiell unterscheiden Chikofsky und Cross bei ihrem Wasserfall Modell drei Phasen, die die Entwicklung der Software durchlaufen muss. Im Falle einer Neuentwicklung steht am Anfang die Requirements Phase, in der die Anforderungen an das System definiert werden, gefolgt von der Design Phase und weiters der konkreten Implementierung. Der Übergang von einer zur anderen

Phase wird über verschiedene Kanten symbolisiert, wobei jede Kante dabei mit einem konkreten Begriff in Zusammenhang gebracht wird. Grundsätzlich gibt es zwei Richtungen, die aus der Sicht der Anforderungen vorwärts oder rückwärts gerichtet sind. Prinzipiell wird daher zwischen Forward Engineering und Reverse Engineering unterschieden, wobei im ersten Fall das fertige System das Ergebnis ist und im zweiten Fall die Ausgangsbasis für das weitere Vorgehen bildet.

Das Modell beschriebt den Lebenszyklus der Software. Bei einem klassischen Forward Engineering wird in der Anfangsphase ein allgemeines Konzept entworfen, das von der konkreten Implementierung unabhängig ist. Je weiter man in den Phasen fortschreitet, umso detaillierter wird die Planung und Implementierung. Jede Erhöhung des Detaillierungsgrades geht mit dem Übergang von einem Abstraktionslevel auf einen anderen einher, dadurch zeichnet sich der forward process aus. So wie eine Blaupause eines Gebäudes nur ein abstraktes Modell bildet und je nach Planung und Realisierung unterschiedlich detailliert ausgelegt wird. Dabei gibt es Pläne für das Gebäude, die in einer frühen Phase entworfen wurden und sehr allgemein gehalten sind. Die Pläne für Heizung, die elektrischen Installationen und die Inneneinrichtung sind hingegen erst zu einem viel späteren Zeitpunkt relevant. Bei der Software stehen am Anfang die Requirements, die in einem späteren, detaillierteren Level mit der konkreten Implementierung zur Deckung gebracht werden müssen.

In diesem gesamten Vorgang muss klar zwischen Levels of abstraction, die über die verschiedenen Stufen der Konzeption reichen und degrees of abstraction, die nur innerhalb einer Stufe vorhanden sind, unterschieden werden. Betrachtet man einen life-cycle, wird in den einzelnen Phasen der Übergang von einem höheren abstraction Level in den Anfangsphasen zu einem niedrigeren Level in

den Endphasen vollzogen. Hingegen können Informationen in einer Stufe immer in einem höheren Abstrahierungsgrad dargestellt werden, wenn sie eine allgemeine, globale Sicht bieten oder in einem niedrigen Abstrahierungsgrad, wenn sie eine detailierte Sicht eines bestimmten Bereichs bieten.

### 3.1.2 Definitionen

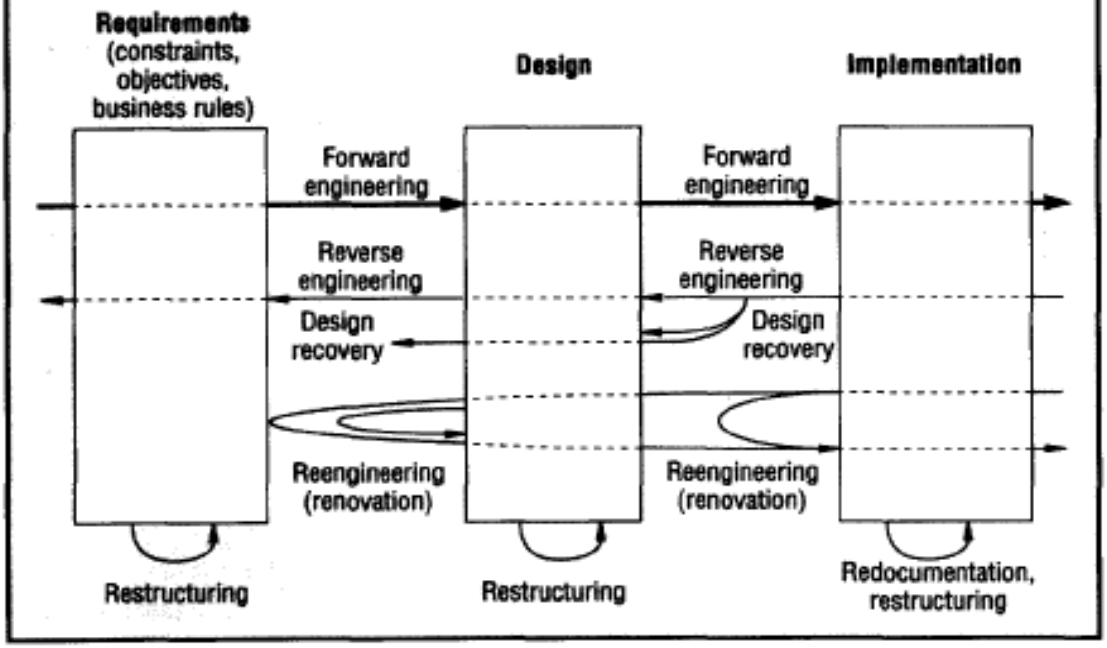


**Figure 1.** Relationship between terms. Reverse engineering and related processes are transformations between or within abstraction levels, represented here in terms of life-cycle phases.

**Abbildung 19: Reverseenigeering, Begriffe und Beziehungen[15]**

In der Grafik werden die drei life-cycle Stufen aufgezeigt. In der Requirements Phase wird die Spezifikation erstellt, die das Problem, das gelöst werden soll beschreibt. Das beinhaltet alle Ziele, Geschäftsvorgänge und deren Beschränkung. In der Design Phase wird die Lösung des Problems spezifiziert. Und die Implementierung umfasst schließlich die Bereiche Programmierung, Testen und die Auslieferung des Systems. Im Folgenden werden die einzelnen

Begriffe, die den Übergang von einer Phase zur anderen bezeichnen, erläutert.

**Forward Engineering**

Dies ist der traditionelle Prozess, bei dem von einem hohen Abstraktionsgrad, von logischen, von der Implementierung unabhängigen Designs, eine physikalische Implementierung des Systems erreicht wird. Der Ausdruck Forward wird hier vor allem deshalb eingeführt, um eine klare Unterscheidung von dem umgekehrten Prozess, dem Reverse Engineering zu erreichen.

**Reverse Engineering**

Dies ist der Prozess, der durchgeführt werden muss, um die Komponenten zu identifizieren und um weiters eine Systembeschreibung auf einem höheren Abstraktionslevel zu erhalten. Es wird versucht, Design und Aufbau des Systems auf einer von der Implementierung unabhängigeren Art und Weise zu erzeugen. Bei dem Vorgang des Reverse Engineering kann man von jedem Punkt des life-cycle starten und versuchen, einen höheren level of abstraction zu erreichen. Dabei wird das System nicht verändert oder erneuert, sondern lediglich untersucht.

Betrachtet man nun den gesamten life cycle, so startet Reverse Engineering bei der konkreten Implementierung und versucht, das Design wieder herzustellen und daraus wiederum die Requirements zu definieren.

Innerhalb des Reverse Engineering findet man viele Teildisziplinen – die zwei wichtigsten sind Redocumentation und Design Recovery.

Beim Vorgang der Redocumentation wird versucht, eine semantisch äquivalente Dokumentation auf demselben Abstraktionslevel wieder herzustellen. Das Ergebnis sind unterschiedliche Darstellungen, z. B. in Form von verschiedenen UML Diagrammtypen oder ER

Diagrammen, mit dem Ziel, eine verständliche Sicht zu präsentieren, die vom Menschen leichter erfasst werden kann. Redocumentation ist die einfachste und älteste Form des Reverse Engineering. Das „re-„ in dieser Bezeichnung impliziert, dass es einmal eine Dokumentation gegeben haben muss und diese nun wieder hergestellt werden soll. Es gibt einige Tools, die diese Aufgabe unterstützen wie *pretty printers*, die den Code in einer leserlichen Form (z. B. : Syntax Highlighting) darstellen oder Diagrammgeneratoren, die Control Flows und Datenstruktur aus dem Code wieder sichtbar machen. Eine Schlüsselaufgabe dieser Tools besteht darin, auf eine einfache Art und Weise die Relationen unter den Komponenten zu analysieren und aufzuzeigen.

Beim Design Recovery**,** das ebenfalls ein Teilvorgang des Reverse Engineering ist, wird zusätzlich das Domain Wissen zur Beschreibung hinzugefügt, des weiteren externe Informationen über die Systemgrenzen hinaus, sowie unscharfe und ungenaue Zusammenhänge. Dadurch kann man mögliche Zusammenhänge auf einem höheren Abstraktionsgrad erkennen und aufzeigen. So entstehen Beschreibungen des Systems auf einem höheren Abstraktionsgrad, die aus einem Konglomerat aus dem Code, noch bestehenden Design Informationen, persönlicher Erfahrung und generellem Wissen über die Domäne bestehen. Design Recovery soll alle Informationen wieder herstellen die eine Person benötigt, um das System und seine Abläufe zu verstehen.

**Restructuring**

Restructuring ist die Transformation von einer Repräsentation in eine andere im gleichen Abstraktionsgrad, wobei Funktionalität und Semantik gleich bleiben müssen. Bezieht man diesen Begriff auf Programmcode spricht man auch häufig von Refactoring[16]. dem Refactoring .Das Ziel dabei ist eine verbesserte Struktur des Designs.

Eine weit verbreitete Form des Restructuring ist die Veränderung des Programmcodes. Dabei wird eine code-to-code Transformation durchgeführt. Aus einem überlangen, zusammenhängenden Programmcode (Spaghetticode) wird durch Umstrukturierung ein oder mehrere gut strukturiere (go-to-less), äquivalente Codesegmente. Eine ähnliche Methode ist die data-to-data Restrukturierung. Dabei werden die Datensätze und Relationen analysiert und normalisiert, um Datensätze, die mehrmals vorkommen, zu eliminieren. Grundsätzlich wird versucht, ähnliche Datenstrukturen zu erkennen und zu vereinheitlichen, mit dem Ziel, die Modelle klarer und die Datenstrukturen einfacher zu gestalten.

Viele Arten des Restructuring können allein mit dem Wissen über die Struktur durchgeführt werden, ganz ohne Wissen über die Domäne, in der das System eingebettet ist. Z.B. können eine Reihe von if Programmstrukturen durch eine case Anweisung ersetzt werden, ohne zu wissen, welche Funktionalität der Domäne sie abbilden.

Restructuring verändert das System und erzeugt neue Versionen. Die Funktionalität und Semantik soll dabei aber unverändert bleiben. Es werden keine Veränderungen aufgrund neuer Anforderungen aufgenommen.

Daher ist Restructuring oft auch eine Form präventiver Wartung, da so versucht wird, die Struktur des Systems und somit auch die Qualität zu verbessern oder zumindest zu erhalten.

**Reengineering**

Reengineering, auch bekannt unter den Begriffen Renovation und Reclamation, geht über das Restructuring hinaus. Das bestehende System wird analysiert und verändert, das Ergebnis des Prozesses soll ein neues System sein, das den veränderten Anforderungen gerecht wird. Reengineering beinhaltet zum einen Reverse Engineering, um eine abstraktere Sicht des Systems zu erhalten,

gefolgt von einem Forward Engineering oder Restructuring, was auch neue Anforderungen beinhalten kann, die nicht Teil des alten Systems waren. Zum Beispiel wird im Zuge des Reengineering Prozesses festgestellt, welche Rollen und Geschäftsprozesse das System abbildet. Dabei werden Veränderungen an den Prozessen durchgeführt, um die Abläufe zu optimieren. Dadurch verändern sich die Anforderungen, die neu in das System aufgenommen werden müssen. Bereits implementierte Anforderungen verlieren möglicherweise ihre Gültigkeit und müssen aus dem System entfernt werden, um es nicht unnötig zu vergrößern. Dabei werden grundlegende Strukturen verändert, also das System auf Basis dieser neuen Anforderungen umgebaut. Dabei soll gleichzeitig die Qualität des Systems, der Struktur und des Designs verbessert werden. Obwohl Reengineering die Techniken des Forward sowie des Reverse Engineering verwendet, besteht es nicht aus den beiden. Vielmehr macht es Gebrauch der beiden Technologien, die sich beide für sich weiterentwickeln, ohne dass sich am Prinzip des Reengineering etwas ändert.

### 3.1.3 Ziele des Reverse Engineering

Was will man nun eigentlich mit Reverse Engineering erreichen? Das primäre Ziel ist es, ein Verständnis dafür zu erhalten, welche Auswirkungen und welchen Umfang die Wartung oder die Neuentwicklung des Systems haben. Es gibt hierfür sechs Teilziele, die mit diesem Prozess erreicht werden sollen.

**Die Komplexität bewältigen.**

Es müssen Methoden gefunden oder entwickelt werden, die besser mit der Größe und Komplexität des Systems zurecht kommen. Ein Schlüssel, um diese Ziele zu erreichen, ist die Automatisierung der

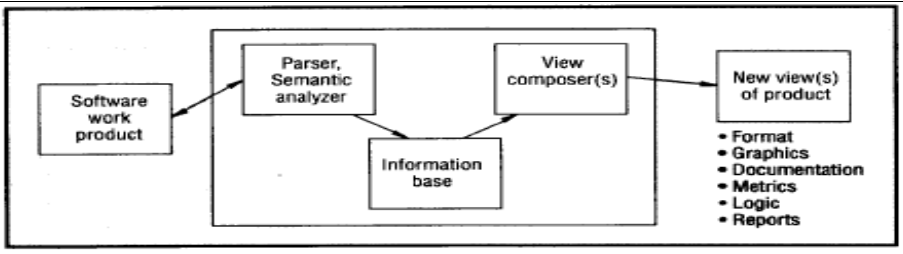


**Figure 2.** Model of tools architecture. Most tools for reverse engineering, restructuring, and reengineering use the same basic architecture. The new views on the right may themselves be software work products, which are shown on the left. (Model provided by Robert Arnold of the Software Productivity Consortium.)

**Abbildung 20: Tools und Methoden[15]**

Vorgehensweise. Verschiedene Tools und Methoden, vereint in einer CASE Umgebung, helfen bei der Aufgabe, die relevanten Informationen aus dem System zu extrahieren und Entscheidungen für die Zukunft bereit zu stellen. Die folgende Grafik zeigt, wie die meisten CASE Tools aufgebaut sind:

**Alternative Ansichten generieren**

Grafische Ansichten erleichtern das Verständnis über die Abläufe ungemein und haben sich schon lange etabliert. Früher verwendete man flow Charts und Data-flow Diagramme, heute sind diese den verschiedenen UML Diagrammtypen gewichen. Die Diagramme, die zu einem Zeitpunkt erzeugt wurden, müssen im Lauf der Zeit auf dem aktuellen Stand gehalten werden. Das ist vielfach nicht der Fall und stellt so einen Flaschenhals im Prozess dar. Abhilfe kann man durch CASE Tools schaffen, die automatisiert aktuelle Ansichten aus dem vorhandenem Programmcode generieren. Dabei müssen diese nicht zwangsläufig grafischer Natur sein, sie können auch auf Text basieren z. B. Javadoc. Viele Entwickler arbeiten nur auf einer Ansicht, doch um Reviews und Audits durchzuführen, sind mehrere Ansichten von Vorteil, auch hier liegt ein entschiedener Vorteil in der automatisierten Erstellung der Dokumentation.

**Verlorene Informationen wiederherstellen**

Bei großen, langlebigen Systemen, die ständig verändert und angepasst werden, geht über die Zeit ein Großteil der Information über das Design verloren. Die meisten Veränderungen werden meistens nicht in die Dokumentation übernommen, somit ist diese veraltet und entspricht nicht mehr dem aktuellen Stand. Das betrifft vor allem jene Bereiche der Dokumentation, die einen höheren Abstraktionsgrad haben. Daher wird mit Reverse Engineering, im Speziellen mit Redocumentation versucht, die Informationen auf einem höheren Abstraktionsgrad wieder herzustellen und dadurch ein besseres Verständnis zu erlangen, welche Systemkomponenten vorhanden sind und wie diese zusammenarbeiten.

**Nebeneffekte aufspüren**

Planloses Design am Anfang des Projekts und sukzessive Änderungen am laufenden System können zu Verästelungen führen, die Nebeneffekte hervorrufen oder die Performance auf viele Arten negativ beeinflussen. Mit Hilfe von Reverse Engineering kann man derartige Nebeneffekte beobachten und analysieren und bei einem Forward Engineering berücksichtigen. Dadurch können Anomalien und Probleme aufgespürt werden, bevor BenutzerInnen diese später als Bug melden.

**Höhere Abstraktionen synthetisieren**

Durch Reverse Engineering versucht man durch die verschiedenen Techniken und Methoden Dokumente und Ansichten des Systems zu erstellen, die auf einem höheren Abstraktionsgrad angesiedelt ist. In welchem Ausmaß dies automatisiert durch Werkzeuge erfolgen kann, hängt vom jeweiligen System ab.

**Vereinfachte Wiederverwendung**

Im Bezug auf die Software Wiederverwendung stellt der große Umfang des bestehenden Systems ein klares Hindernis dar. Durch

Reverse Engineering werden wiederverwendbare Teile des Systems identifiziert, aus denen dann wiederverwendbare Komponenten erzeugt werden können.

## 3.2 Legacy Systems

Seit 30 Jahren wird Software Engineering als Ingenieurwissenschaft vorangetrieben und macht dabei beachtliche Fortschritte. Heute wird Software mit anderen Mitteln entwickelt als noch von 10, 20 oder 30 Jahren. Relationale Datenbanken gelten heute als Selbstverständlichkeit, Objektorientierung und UML sind weiter verbreitet. Es muss schon gute Gründe dafür geben, um Programme heute noch in Assembler, Cobol oder Modula zu schreiben. Ein Großteil der Software Ingenieure setzt verständlicher Weise auf State of the Art Techniken. Viele Programme wurden damals ohne konkretes Ablaufdatum geschrieben, sie haben aber keinen klassischen Verschleiß, wie wir es von anderen Produkten kennen. Sie wurden für die nächsten Jahre konzipiert, waren aber oftmals Jahrzehnte im Einsatz. In vielen Bereichen sind wir von derartigen Systemen umgeben, doch uns fällt ihre Anwesenheit erst in dem Moment auf, in dem sie fehlerhaft werden oder gar nicht mehr funktionieren – man erinnere sich an das Jahr Y2K Spektakel... Vielfach gelangen solche Systeme kleinlaut an die Grenze ihrer Leistungsfähigkeit, weil z. B. ein Entwickler damals der Meinung war, 10.000 Einträge sind mehr als genug. Doch wird nach vielen Jahren dieser 10.000. Eintrag eingegeben, kann eine Routinetätigkeit schnell zu einem Canossagang ausarten.

Viele Unternehmen sind von derartigen Systemen abhängig, oft sind sie unentbehrlich für die grundlegenden Geschäftsvorgänge. Das

Risiko, dass ein derartiges System fehlerhaft arbeitet, ist bekannt und die Budgets der Systeme werden permanent gesprengt, da genau diese unerwarteten Fehler schnell behoben werden müssen. Typischerweise versickern daher große Summen, nur um den Betrieb aufrecht zu erhalten und das Unternehmen nicht zu gefährden.

Unternehmen entwickeln sich über die Jahre weiter. Die Informationssysteme müssen mit dieser Entwicklung schritthalten, es wird verlangt, dass die Systeme ständig den neuen Anforderungen angepasst werden. Die Evolution der Software muss mit der Evolution des Unternehmen synchronisiert werden. Durch die Evolution muss gewährleistet werden, dass ein adäquater Qualitätslevel aufrecht erhalten bleibt, damit die Systeme einfach gewartet werden können. Leider sieht es in der Realität anders aus, Änderungen werden oft schnell in Form eines Hot Fix durchgeführt. System Design und Architektur werden dadurch aufgebrochen und Folge dessen leidet die Qualität des Systems als Ganzes. Derartige Systeme werden daher als Legacy Systems bezeichnet.

Bennet [17] beschreibt einige nützliche Symptome, um potentielle Legacy Systeme zu identifizieren: Das Alter beträgt über 10 Jahre, die Größe mehr als 100 LOC, alte Programmiersprachen werden verwendet, eine unüberschaubare Liste mit unzähligen Änderungen, die am System durchgeführt wurden, erhöhen die Entropie und erschweren somit anstehende Wartungsarbeiten. Diese Systeme sind Mission kritisch und tragen ein umfangreiches Fachwissen in sich. Nicht alle diese Punkte müssen für ein Legacy System erfüllt sein. Anzumerken ist auch, dass einige Systeme bereits zu dem Zeitpunkt ihrer Fertigstellung das Attribut Legacy erhalten, da eine viel zu schwache Technologie gewählt wurde.

Um den Term Legacy System auf den Punkt zu bringen, kann man sich der ersten zwei von Lehmans Gesetzen zur Software Evolution

[18] bedienen. Das erste besagt, dass ein Software System, das in Verwendung steht, ständig angepasst werden muss, damit es auch weiterhin zufriedenstellend verwendet werden kann. Das zweite besagt, dass die Evolution eines Systems immer mit einer gesteigerten Komplexität einhergeht, wenn keine Maßnahmen ergriffen werden, um diese Wartungsarbeiten zu steuern und die steigende Komplexität zu mindern.

Aufgrund dieser Feststellung kann man ein Legacy System wie folgt definieren: Ein für die Mission kritisches Software System, das irgendwann in der Vergangenheit entwickelt wurde, von dem wir umgeben sind und das über einen langen Zeitraum verändert wurde, ohne dass diese Änderungen systematisch in das System eingepflegt und Maßnahmen zur Verjüngung durchgeführt wurden [19].

## 3.3 Umgang mit Legacy Systems

### 3.3.1 Strategien für die Zukunft

Um die Entscheidung für den richtigen Umgang mit derartigen Systemen zu treffen, bedarf es einer Klassifizierung [20]. Bennett unterscheidet dabei sechs verschiedene Strategien, wie der weitere Umgang erfolgen soll.

**Verwerfen – Discard**
Die gesamte Software wird verworfen und das System von Grund auf neu entwickelt. Das beseitigt die Legacy Problematik mit einem Schlag, hat aber auch seine Schattenseiten. In das alte System wurde viel Wissen einprogrammiert, das somit verloren geht. Es müssen erst mühsam neue Anforderungen erhoben werden, ein

Design entworfen und das System implementiert und getestet werden. Vor allem die seltenen Ausnahmen, die das alte System abbildet, werden dabei nicht vollständig erfasst werden können und erst im laufenden Betrieb zu Tage treten. Es wird also eine Zeit dauern, bis das neue System zufriedenstellend arbeitet. Das alte System läuft bereits stabil, ist integriert in das Unternehmen und wurde über lange Zeit von großen und kleinen Fehlern befreit, ansonsten hätte es einen so langen Zeitraum nicht überlebt. Das neue System muss diesem Anspruch erst gerecht werden.

**Einpacken – Wrap**

Die einzelnen Komponenten des alten Systems müssen bei dieser Methode zuerst identifiziert werden. Sie werden dann eingepackt oder umwickelt, d. h. mit Schnittstellen versehen, sodass mit einem State of the art System darauf zugegriffen werden kann. Die Applikation und vor allem ihre Business Logik werden dabei nicht verändert. Das System erscheint nach außen in neuen Kleidern, sein Kern ist aber der gleiche geblieben. Eine häufige Ausprägung dieser Vorgangsweise ist das Verpacken von alten Systemen in eine objektorientierte Struktur. Das System wird dabei in einer objektorientierten Struktur gekapselt. Der Zugriff erfolgt dann über die Objekte und nicht mehr direkt auf das alte System. Dabei wird aber nicht das eigentliche Problem beseitigt, sondern nur alter Wein in neue Schläuche gegossen. Die Analogie ist insofern zutreffend, als dass alter Wein oft besser ist als frischer Traubensaft, er kann aber ebenso leicht auch verdorben und somit ungenießbar sein. Trotzdem ist es mit dieser Methodik möglich, die einzelnen veralteten Teile, die identifiziert wurden und nun über Schnittstellen verfügen, nach und nach vollständig zu ersetzen.

**Auslagern – Outsource**

Das Unternehmen entscheidet, dass die Wartung und Pflege von Softwaresystemen, speziell von Legacy Systems, nicht zum Kerngeschäft gehört. Somit werden diese Aufgaben ausgelagert an ein außenstehendes Unternehmen, das diesen Service übernimmt. Diese Vorgehensweise verlagert die Problematik nur, packt sie aber nicht an ihren Wurzeln.

**Einfrieren – Freeze**

In diesem Fall wird die Entscheidung getroffen, dass am System keine Wartungsarbeiten durchgeführt werden. Das System wird so weitergeführt, wie es ist. Dies kann aber nur eine Übergangslösung darstellen.

**Carry On**

Trotz aller Probleme wird entschieden, das System weiter zu führen, allfällige Wartungsarbeiten und Anpassungen werden vorgenommen. Die Entscheidung wird auf eine bestimmte Periode beschränkt.

**Reverse Engineering**

Die Wartung derartig großer Softwaresysteme wird von den Anforderungen, die neu gestellt werden, gesteuert. Grundsätzlich kann man hier von drei Schritten sprechen, um diese neuen oder veränderten Anforderungen in das System zu integrieren.

In einem ersten Schritt müssen Funktionsweise und Prozesse, die das System abbildet, erfasst werden. So ist auch auch die Reaktion des Systems ersichtlich, wenn eine neue Änderung durchgeführt werden soll.

Im zweiten Schritt wird die konkrete Änderung am System vorgenommen. Dabei muss darauf geachtet werden, dass keine Nebeneffekte erzeugt werden, durch die bereits abgebildete

Anforderungen verändert werden oder das System womöglich instabil wird.

Der dritte und letzte Schritt beinhaltet eine Validierung des Systems, wodurch die vorgenommene Veränderung überprüft und getestet und auf auftretende Nebeneffekte geachtet wird.

Begriffsbildung und Definitionen in Bezug auf Reverse Engineering wurden bereits erklärt. Die verschiedenen Ausprägungen können mehr oder weniger radikal ausfallen und das System entsprechend stark oder weniger stark verändern. Grundlegend kann man dazu drei klare Aussagen treffen:

1. Die Entscheidung, was mit einer bestehenden Software passieren soll, wird meist von Ingenieuren angestoßen, in der Hoffnung, eine Tool-basierende Lösung zu finden. Erkennt man aber dann Größe und Umfang des Projektes, so wird klar, dass man alle Stakeholder miteinbeziehen muss, um mit ihnen gemeinsam eine Entscheidung auszuarbeiten.
2. Es eröffnen sich viele verschiedene Möglichkeiten, um die Problemstellung oder Teile davon zu lösen. Jede hat ihre Vor- und Nachteile und bringt so unterschiedliche Risiken mit sich.
3. Auch wenn das Problem technischer Natur ist, gibt es kaum Entscheidungsmodelle und Investment Strategien, wie Unternehmen mit ihren Legacy Systemen verfahren können.

Dazu ist auch zu sagen, dass nur wenige Erfahrungsberichte publiziert werden, das kann mehrere Gründe haben: Zum einen ist der Zeitraum, in dem Veränderungen an Legacy Systemen vorgenommen werden, meist sehr groß, zum anderen besteht die Lösung oft nur aus einer technischen Sicht und drängt die Tatsache in den Hintergrund, dass die Entscheidung Auswirkungen auf das gesamte Unternehmen hat und daher auch von strategischer Bedeutung ist. Integration in die bestehende Planung wird oft

vernachlässigt, vielmehr noch die Bedeutung einer Integration gar nicht wahr genommen. Das sogenannte SABA Projekt (Software As a Business Asset) greift diese Problematik auf. Es basiert auf der Beobachtung, dass sich die Entscheidungsfindung beim Umgang mit Legacy Systemen nicht nur auf einer technischen Ebene bewegt, sondern im gleichen Ausmaß die verschiedenen Management Hierarchien betrifft.

### 3.3.2 Das SABA model

Das SABA Modell ist ein iterativer Prozess, der grundsätzlich aus 2 Stufen gebildet wird. In der ersten wird der Status quo des Systems ermittelt. Dabei wird nicht nur das System an sich betrachtet, sondern auch die Interaktion mit der Umgebung, in der es eingebettet ist. Das erfolgt durch die Verwendung eines Organisational Scenario Tool (OST) [21]. Der Ausdruck Tool ist in diesem Zusammenhang gleichzusetzen mit dem Wort *Methode*.

Die Absicht des OST ist es, den Zweck und das Ziel einer Organisation aufgrund ihres Leitfadens und ihrer Vision zu bestimmen. Daraus entstehen dann ein oder mehrere Szenarien für die Zukunft des Unternehmens. Die Vorgehensweise ist ähnlich einem Business Process Reengineering. Die entstandenen Szenarien bilden dann die Grundlage für das Technology Scenario Tool (TST). Innerhalb dieses Schrittes werden die Legacy Software, Personal und bestehende Prozesse analysiert und eine Lösungsstrategie erarbeitet, die sich mit Zweck und Ziel aus der ersten Phase deckt. Diese technische Lösung wird dann wieder an das OST weitergereicht und dort aufgrund der neuen Erkenntnisse analysiert. Ist man mit dieser Lösung zufrieden, kann dieses Szenario endgültig berechnet werden, ansonsten sind

weitere Iterationen nötig, bis man zu einem Szenario gelangt, das sich mit den Zielen des Unternehmens deckt.

Der wichtigste Aspekt dieses Ansatzes besteht darin, dass er interdisziplinär und von den Unternehmenszielen gesteuert ist. Die Entscheidung wird nicht auf rein technischer Basis gebildet, sondern auch unter Berücksichtigung der ermittelten Unternehmensziele. In der Software Engineering Community liegt der Schwerpunkt bei Legacy Systemen in der Software. Hier wird die Industrie aber nicht ausreichend unterstützt. Man kann diesen Mangel nicht allein damit begründen, dass der Technologiewandel hier nicht so schnell vollzogen werden kann oder dass es eine Reihe erfolgreicher Projekte gibt, bei denen die Legacy Problematik durchaus erfolgreich gelöst wurde. Vielmehr ist die Kritik darauf gerichtet, dass das Problem als rein technischer Natur betrachtet wird und auch die Lösungsansätze entsprechend angesiedelt werden, die unternehmerische Komponente mit ihren Strategien und Zielen jedoch viel zu stark vernachlässigt wird. Wird. Diese technischen Lösungen werden in der Industrie oft kritisiert, da sie risikoreich, unerprobt, finanziell kostspielig, personalintensiv, in bestehende Vorhaben nicht integriert sind und jegliche Kosten-Nutzen-Analyse fehlt. Sicherlich treffen nicht alle Probleme gleichzeitig zu, die Vorbehalte sind aber berechtigt. Typischer Weise folgen die Legacy Systeme nicht schnell genug den Veränderungen, denen das Unternehmen unterworfen wird und die Integration in die Unternehmensprozesse ist meist schwach. Der SABA-Ansatz berücksichtigt im Gegensatz dazu beide Seiten, die technische sowie jene innerhalb der Organisation.

Die Informationen, auf denen die Entscheidungen schlussendlich beruhen sollen, müssen in einer Art aufgearbeitet werden, dass sie dem jeweiligen Level und der Art der Entscheidung gerecht werden. Sie können individuellen aber auch allgemeinen Charakter haben. Alle Stakeholder der Organisation sind eingeladen, an der Ausarbeitung

dieser Informationen teilzunehmen. Der Entscheidungsprozess wird nicht mehr von unten nach oben gesteuert, wo technisches Personal die Lösung ausarbeitet. Es werden alle Stakeholder, die an der Entscheidung teilnehmen, eingeladen, Informationen bereitzustellen, die für die Entscheidung nützlich bzw. wichtig sind und auf Basis derer dann eine Entscheidung gefasst wird, die auf einer organisatorischen und nicht mehr technischen Ebene stattfindet. Das erfordert aber auch Ressourcen, die von allen Stakeholdern bereitgestellt werden müssen.

Die Ziele des SABA Models sind:

- Es soll eine analytische Methode bereit stellen, die Organisationen behilflich ist, eine Entscheidung zu treffen, wie sie mit ihrem Legacy System umgehen soll.
- Basierend auf dieser Entscheidung wird ein Plan entworfen, wie der Übergang vom derzeitigen System zum neuen erfolgen soll.
- Die iterative Methode kann bereits in einem Zeitraum von 2 Monaten schnelle Ergebnisse liefern, die dann verfeinert und besser approximiert werden können.
- Die benötigten Informationen werden von den verschiedenen Stakeholdern eingeholt und auf verschiedenen Levels aufgearbeitet. So kann eine sachgemäße Entscheidung für das Unternehmen zum Legacy System getroffen werden.
- Das Ergebnis besteht aus einem interdisziplinären Ansatz der Management Theorien, System Informationen und des Software Engineering.
- Es soll ein Framework bereitgestellt werden, das die verschiedenen Technologien im Umgang mit Legacy Systemen (z. B. Reverse Engineering) in dem Maße einsetzt, wie sie benötigt werden.

- Es soll dazu bewegen, die Betrachtungsweise der Software über einen längeren Zeitraum auszudehnen, nämlich als Unterstützung des gesamten Lebenszyklus.
- Es soll eine analytische Methode für Unternehmen bereitstellen, mit der sie zukünftige Entscheidungen bezüglich ihrer Software treffen können.

**Phase1: Organisational Scenario Tool (OST)**

Das Organisational Scenario Tool ist eine Methode, die die verschiedenen Stakeholer auf den verschiedenen Ebenen der Organisation in die Planung einbezieht. Die Methode der Szenarienbildung ist in der Informationstechnologie bereits etabliert und wurde durch das OST weiterentwickelt. Es richtet den Fokus auf die verschiedenen Stakeholder und rückt sie bei der Bewertung der Szenarien ins Zentrum. Der Vorgang wird von einem Moderator gesteuert, der nicht zwangsläufig aus der Organisation stammt. Es soll eine Person sein, die nicht gleichzeitig Stakeholder ist. Sie beginnt mit der Analyse des bestehenden Organisational Scenario und der Bewertung nach neun Kriterien:

1. *boundary*: die zu untersuchende Einheit
2. *vision*: die allgemeine Beschreibung der Einheit
3. *logic*: Gründe für die Vision
4. *structure*: Struktur der Organisation
5. *roles*: organisatorische Rollen der einzelnen Personen
6. *view of information*: Analysieren der Resourcen
7. *costs*: Kosten – finanzielle und nicht finanzielle
8. *benefits*: Aufwendungen – finanzielle und nicht finanzielle
9. *risks*: größte Gefahrenquellen

Die Ergebnisse der Untersuchung werden in eine Tabelle eingetragen. Dabei ist auch eine iterative Vorgehensweise möglich [22]. Danach werden Szenarien an die Stakeholder übergeben, die als Ausgangspunkt für die weiteren Entscheidungen dienen.

Diese Szenarien betreffen die Organisation als Ganzes und nicht nur die reinen technischen Möglichkeiten. Für jedes Szenario wird dann ein technisches Porfolio erstellt, das die eingesetzten Technologien beschreibt, um dieses Szenario zu verwirklichen. Das soll wiederum von den Stakeholdern durchgeführt werden und nicht vom technisch orientierten Personal.

**Phase 2: technology scenario tool**

Ziel dieser Phase ist es, für jedes Szenario die entspechende Technologie für die Umstellung zu finden, das Ergebnis ist somit eine Menge an Möglichkeiten, um ein Szenario zu realisieren.

Das Diagramm unten stellt der Einfachheit halber die Vorgehensweise als Wasserfall Modell dar, es sind aber auch interative Modelle möglich [23].

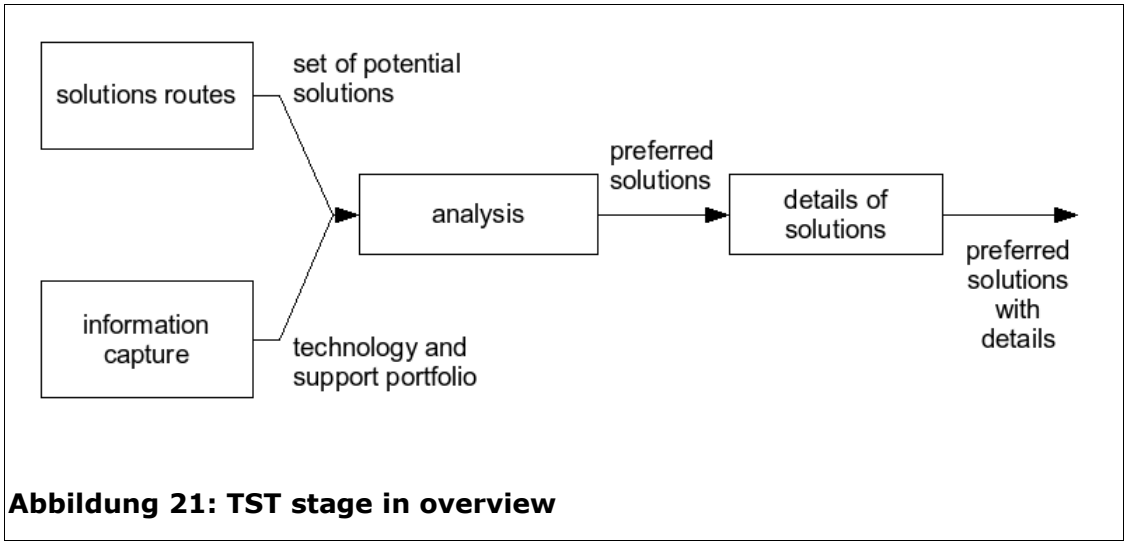


**Abbildung 21: TST stage in overview**

Die erste Tätigkeit ist jene der **Informationserfassung** (information capture). Dabei werden die Ergebnisse des OST analysiert und auf die Realisierung hin überprüft. Als Ergebnis erhält man detaillierte

Informationen, um das Software Engineering durchzuführen, aber auch Sprachen, struktureller Aufbau und die Art und Weise, wie die laufende Instandhaltung später gewährleistet werden soll.

Parallel dazu werden die verschiedenen **Lösungswege** (solutions routes) erstellt. Parallel deshalb, da verschiedene Wege verschiedene Informationen benötigen.

Einige Lösungen können durch das Stellen einfacher Fragen sehr schnell verabschiedet werden, z.B.:

- *Leave*: Wurde die strategische Entscheidung das System zu ersetzen bereits getroffen?
- *Discard*: Ist das System Teil des Kernprozesses? Könnte das Unternehmen in seiner Form weiter bestehen, wenn man auf das System verzichten müsste?
- *„Re-quire" (Aktualisieren der bestehenden Anforderungen und Bau eines neuen Systems.)* Wie groß ist der Aufwand für diesen Weg? Welche Anforderungen bestehen bereits? Wie aktuell sind diese?
- *Re-engineering [24]*: Ist der Quellcode noch vorhanden? Existieren Werkzeuge, um dieses zu analysieren? In welcher Sprache wurde das System geschrieben? Wie groß ist das System (LOC)?

Das Ergebnis der beiden Schritte wird dann im **Analyse** Schritt weiterverarbeitet. Hier sollte eine Entscheidung getroffen werden, welche der möglichen Lösungen verwendet wird. Die Ergebnisse aus den vorangegangenen Schritten bilden dabei die Grundlage, um die EntscheidungsträgerInnen mit Expertenwissen zu versorgen. Das Ergebnis sollte eine engere Auswahl der bestehenden Möglichkeiten darstellen. Diese werden dann beim nächsten Schritt, der **detaillierten Ausarbeitung,** verwendet. Hier werden Zeit, Kosten,

Risiken und Personalaufwand ermittelt. Aber auch die Softwarelösung an sich wird noch einmal genau daraufhin untersucht, ob sie den Erfordernissen schlussendlich entsprechen wird.

**Ergebnis der Phasen**

Schließlich ergeben sich eine Menge von Szenarien vom OST, die die organisatorische Entwicklung des Unternehmens berücksichtigen. Diese Szenarien bzw. eine Teilmenge davon, wird in die technischen Szenario Tools übernommen. So erhält man einen Vergleich der verschiedenen technischen Lösungen und zudem einen Vergleich mit dem Legacy System. Durch Diskussionen auf dieser sachlichen Grundlage wird schlussendlich ein Lösungsweg ausgewählt, der dann von den Softwareentwicklern ausgearbeitet wird.

Diese Vorgehensweise wirkt der Erkenntnis entgegen, dass rein technische Lösungen für Legacy Systeme unzureichend sind.

# 4 Qualitätssicherung in der GKN Sintermetals

Das folgende Kapitel stellt den Konzern, in den diese Arbeit eingebettet ist, kurz vor, beschreibt weiter das Werk und geht im Speziellen auf die Qualitätsicherung in diesem Werk ein.

## 4.1 GKN Unternehmensprofil

GKN stellt Produkte und Ingenieurleistungen bereit, die im Automotive und Aerospace von namhaften Herstellern eingesetzt werden. Es handelt sich um ein global aufgestelltes Unternehmen, das weltweit mehr als 40.000 Mitarbeiter beschäftigt. Es besteht aus über 250 Produktionswerken und Joint Ventures in über 30 Ländern. Die Ursprünge reichen weit in das 18. Jahrhundert zurück. 1759 wird die Dowlais Iron Company in Wales gegründet.

Die Industrialisierung der Britischen Insel und die Eisenbahnmanie greifen um sich und Dowlais stellt im Zuge dessen tausende Kilometer Bahnschienen her und Millionen dazugehörender Bolzen. 1852 übernimmt Lady Charlotte Guest die Führung über Dowlais, die bereits seit 1840 zu einer der größten Eisenhütten weltweit herangewachsen ist.

Im Jahre 1854 beginnt die John Sutton Nettlefold mit der Massenproduktion von Schrauben und lizensiert als Erster das Bessemer Verfahren zur Stahlerzeugung.

Arthur Keen kauft Nettlefold und Dowlais und gründet die Guest, Keen & Nettlefolds Ltd (GKN), sie wird der weltgrößte Hersteller von

Schrauben, Nägeln, Muttern und Bolzen. Das Unternehmen setzte dabei auf eine durchgehende Produktionslinie, die vom Erz- und Kohleabbau, über die Verarbeitung bis zur Vermarktung ging.

Im Zweiten Weltkrieg wird Kriegsmaterial produziert und 1966 erfolgt mit dem Kauf von *Hardy Spicer Limited*, einem Hersteller von Antriebswellen für PKW, die Ausrichtung zu einem Automotiv Zulieferer. In den folgenden Jahrzehnten fanden weitere Übernahmen statt, wodurch GKN zum Weltmarktführer für Antriebsteile wurde, mit einem Anteil von 43 Prozent im Jahr 2002.

Die Kernbereiche sind heute die Produktion von Antriebskomponenten sowie ganzer Bauteilgruppen, die im On- und Offroad Bereich zum Einsatz kommen. In 85 % aller derzeit produzierten Automobile sind Bauteile von GKN zu finden. Weiters werden Komponenten für die Luftfahrtindustrie hergestellt und von namhaften Herstellern, zum Beispiel im Airbus 380, F-35 verbaut.

Mit der Sintertechnologie wurde in den letzten 20 Jahren ein weiterer Kernbereich geschaffen. Der Bereich GKN Sinter Metals ist mit über 35 Produktionsstätten und einem Anteil von 16 Prozent Weltmarktführer bei der Produktion von Bauteilen mit dieser Technologie.

## 4.2 GKN Sintermetals

GKN Sintermetels Bruneck ist eines der weltweit über 30 Werke der GKN, in denen Sinterteile hergestellt werden. Im Vergleich zu anderen Verfahren der Metallverarbeitung ist die Pulvermetallurgie durch eine Reihe außergewöhnlicher, verfahrenspezifischer Eigenheiten geprägt. Ausgehend von Metallpulvern werden in einem mehrstufigen Prozess einbaufertige Bauteile erzeugt.

Für die positive Entwicklung der Pulvermetallurgie gibt es gute wirtschaftliche Gründe: Häufig sind die Herstellungskosten von Bauteilen mit hohen Anforderungen an die Maßgenauigkeit für die pulvermetallurgische Fertigung geringer im Vergleich zu Gieß- oder Spanenden Verfahren.

Der Vorteil ist umso ausgeprägter, desto höher die Anforderung nach gleichbleibender Qualität bei größeren Serien ist. Pulvermetallurgie ist eine der flexibelsten Fertigungstechniken, durch die richtige Werkstoff- bzw. Pulverauswahl können hochporöse Bauteile, hochfeste Bauteile mit isotropen Eigenschaften oder Verbundwerkstoffe hergestellt werden. Durch Kombination mit dem richtigen Formgebungsverfahren können verschiedenste Familien von Bauteilen erzeugt werden.

Das Werk Bruneck ist dabei der Center of Excellence für den Bereich Lagertechnologie.

Gegründet wurde das Werk im Jahre 1967, damals mit dem Namen Bound Brook, dieser stammt von dem gleichnamigen Städtchen im Staate New Jersy an der Ostküste der Vereinigten Staaten.

Schon im Jahre 1921 befasste sich die dort ansässige Bound Brook Bearing Cooperation mit der Herstellung von Sintergleitlagern und den überragenden Vorteilen der Pulvermetallurgie.

Nachfolgend eine kurze Timeline des Unternehmens.

1968 werden in Bruneck die ersten Sintergleitlager gefertigt.

1974 wurde der erste Werkzeugneubau errichtet.

1980 wurden Service Abteilungen und Verwaltung erweitert. Aufbauend auf dieser Grundlage wurden dann im selben Jahr die ersten Sinterformteile Produziert. Die logische Weiterentwicklung der Sintergleitlager-Herstellungstechnik

| | |
|---|---|
| | hat unser Fertigungsprogramm sinnvoll ergänzt. |
| 1984 | Erweiterung der Fertigungsmassnahmen. |
| 1991 | Übernahme des Sinterwerks GKN SAINI Spa in Cernusco s/ N Mailand mit entsprechender erweiterung der Produktpalette mit einem Pressbereich bis 500t. |
| 1998 | Namenänderung auf GKN SINTER METALS AG/SPA |
| 2000 | Beginn mit der Produktion von Kolben und Pleuel für Kompressoren. |
| 2004 | Errichtung einer neuen Halle für Werkzeugbau |

**Tabelle 1: Zeitachse Gkn Sintermetals**

## 4.3 Eckdaten des Unternehmens

Anhand einiger Kennzahlen wird hier das Unternehmen kurz vorgestellt, diese Kennzahlen können leicht mit anderen Unternehmen verglichen werden.

### 4.3.1 Allgemeine Informationen

Folgende Informationen geben einen kurzen Überblick über die Grösse des Unternehmens:

- Fläche vom Werksgeländes 21.985 m²
- Umsatz EURO 70.6 M (2006)
- Mitarbeiter 452

### 4.3.2 Qualitätszertifikate

GKN Sintermetals Bruneck ist nach folgenden Qualitätsnormen zertifiziert:

- ISO 9001:2000
- ISO/TS: 16949:2002
- ISO 14001
- Q1 (Ford)
- Gold certificate of Recognition (Denso)

### 4.3.3 Produktionsanlagen

GKN Sintermetels besitzt ein umfangreiches Portfolio an Maschinen und Anlagen in der Produktion. Nachfolgend eine kurze Auflistung der verschiedenen Einheiten, sie ermöglichen eine Vorstellung von der Größe und vom Umfang der Produktion.

132 Pulverpressen von 4 bis 100 t

40 Kalibrierpressen

14 Sinteröfen

5 Tränkstationen

6 Härteöfen

3 Dampfhärteöfen

2 Sinterhärteöfen

## 4.4 Unternehmensstruktur

Das Organigramm des Managements spiegelt im wesentlichen die Unternehmensstruktur wider. Dabei wird dieser hierarchische Aufbau durch eine global übergeordnete Matrixorganisation ergänzt, d. h.

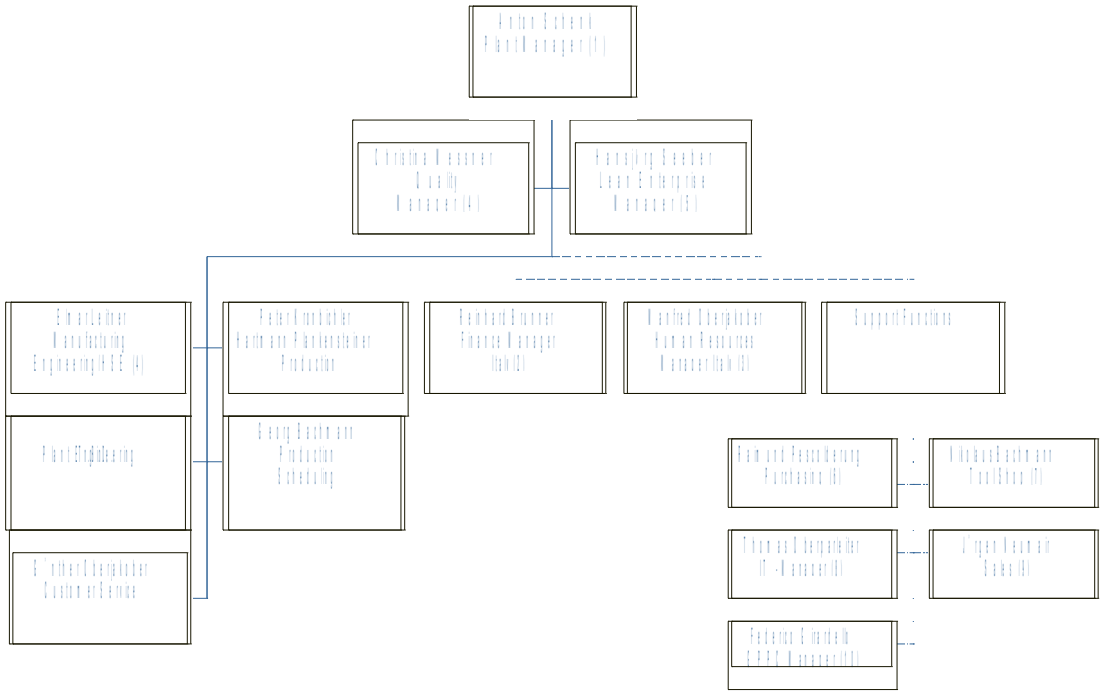

Abbildung 22: Organigramm Sintermetals Bruneck

jeder Abteilungsleiter leistet ein zusätzliches Reporting an eine übergeordnete Stelle.

**Matrix Organization Reporting:**

(2) Director Quality / HSE Europe

(3) Lean Enterprise Europe

(4) Director Global Purchasing

(5) Vice President Engineering Europe

(6) Director Global Information Technology

(7) Director Sales KA Group 3

(8) VP GPPC Small Segment

**Reporting to:**

(2) Vice President Operations (P. Oberparleiter)

(3) Director Finance & Controlling

(4) Human Resources Director Europe

Die Qualitätsabteilung ist hier als Stabstelle angelegt und umfasst ca. 20 MitarbeiterInnen. Diese sind z. T. in der Wareneingangs-, vor allem aber in der Warenausgangskontrolle und in der Fertigung an den verschiedenen SPC Stationen tätig.

## 4.5 Entwicklung des Qualitätswesens

Die Entwicklung der Qualitätsabteilung bzw. des Qualitätsgedankens im Unternehmen hat verschiedene Stufen durchlaufen. Dabei wurde durch die unten angeführten Maßnahmen eine stetige Performanceverbesserung erzielt.

| | |
|---|---|
| 1987 | Start eines strukturierten Qualitätsmanagement Systems durch Reengineering der Organisation und flächendeckender Qualitätsintegration |
| 1992 | Erste Zertifizierung nach ISO 9002 |
| 1997 | Zertifizierung der Automotive Standard Anforderungen QS-9000 und VDA 6.1 |
| 2003 | Zertifizierung nach ISO/TS 16949 und ISO 14001 |
| der-zeit | Lean Enterprise<br>*Quality HSE Standards<br>*Abfallvermeidung und -management<br>*Präventives Fehlermanagement<br>*Kontinuierliche Verbesserung |

**Tabelle 2: Zeitachse Entwicklung des Qualitätswesens**

## 4.6 Ausgangsbasis und Geschichte der Qualitätssoftware

Das CAQ System von GKN Sintermetals Bruneck wurde Ende der 80er Jahre, in Zusammenarbeit mit der University of Aarhus, im Zuge einer Neuorganisation des Managementsystems, entwickelt. Das System wurde über die Jahre angepasst und optimiert. So wie jedes System mit der Zeit altert, ist auch dieses mit Technologien entwickelt worden, die heute nicht mehr state of the art sind. Diese Systeme werden allgemein als Legacy Systeme bezeichnet – als Systeme, die in die Jahre gekommen sind.

Das CAQ System bildet Geschäftsprozesse ab, die zum Teil kritisch sind. Ein Aspekt, der oft vergessen wird, ist, dass hier über die Jahre viel Know-how hineinprogrammiert wurde und dieses System somit zum Wissensträger geworden ist. Nun soll CAQ Bruneck modernisiert werden. Dabei gelten Vorgaben, aus denen sich 2 Szenarien ableiten lassen:

- Einführung des CASQ-IT 9000 Softwarepakets von B&W oder
- Reengineering des bestehenden Systems

## 4.7 Systemgrenzen und technische Architektur

Das System betrifft nur das System „CAQ Bruneck" und die Migration auf ein äquivalentes System. Folgende Module sind davon nicht betroffen:

- FMEA
- APQP

- Prüfmittelmanagement
- Prüfmittelfähigkeit

Im Kern besteht das System aus der Prüfplanung, der SPC sowie einem Management System für interne Abweichungen. Hierzu gehören jedoch nicht Kundenreklamationen, für diese Art wird ein eigenes System verwendet.

Zudem ist die MDE in das System integriert und stark an die SPC gekoppelt. D. h. die Clients werden verwendet, um zusätzlich zu den Qualitätsdaten auch Maschinendaten oder allgemeine Betriebsdaten zu erfassen. Über ein zentrales Auswertungstool kann am Controlstand ein Überblick über alle Maschinen angezeigt werden sowie Detaildaten für jede Maschine. Die Auswertung ist nicht Teil des Systems CAQ Bruneck, wohl aber werden die MDE Daten gemeinsam mit den SPC Daten angezeigt und ausgewertet, so können Rückschlüsse gezogen werden: Wurde z. B. ein Werkzeugwechsel durchgeführt und danach treten Abweichungen auf, können diese später unmittelbar in Zusammenhang gebracht werden.

Bei diesem System wurde ansatzweise das Client – Server Prinzip umgesetzt. Wobei der Server in diesem Falle durch einen einfachen Dateiserver repräsentiert wird. Die Clients sind vorwiegend Programme, die mit Turbo Pascal geschrieben wurden und auf einem MS-DOS Betriebssystem laufen. Und das vorwiegend an den Messstationen oder in einer MS-DOS Umgebung der verschiedenen Microsoft Betriebsystemen an den Arbeitsplätzen in den Verwaltungsbüros.

Da die Daten in keiner Relationalen Datenbank abgelegt wurden, kommt es immer wieder zu Kollisionen, da mehrere Clients gleichzeitig das selbe File beschreiben. Die Folgen sind Datenverlust oder auch Integritätsverletzungen der Daten, wenn meherere Dateien gleichzeitig beschrieben werden.

Auch muss man dem hinzufügen dass die Daten nicht als Text bzw Ascii files abgeleget werden sondern in einem proprietätem Binärformat, das das Ziel hatte den Speicherplatz klein zu halten. So passt heute die gesamte Datenmenge die in einem Jahr anfällt locker auf eine CD-ROM Disk.

## 4.8 CAQ Module Ist-Erhebung

CAQ Bruneck besteht derzeit aus 3 Modulen, die auf einer gemeinsamen Datenbasis operieren. Vor allem das Modul SPC83 ist sehr umfangreich und deckt Bereiche ab, die über die statistische Prozesskontrolle weit hinausreichen.

Z.B. ist im System Bruneck auch die MDE integriert, damit ist unter anderem eine kompakte Arbeitsweise und eine schnelle und lückenlose Rückverfolgung aller SPC und Prozessdaten gegeben.

Alle Module wurden mit Turbo Pascal entwickelt und enthalten in Summe ca. 16.000 Programmzeilen. Die Daten werden in Dateien gespeichert – und das ist auch der große Schwachpunkt des gesamten Systems. Auch sind die eingesetzten Technologien MS-DOS und Turbo Pascal nicht mehr zeitgemäß und es kam in der Vergangenheit häufig zu Störungen im Netzwerk durch gleichzeitigen Zugriff verschiedener Clients auf eine gemeinsame Resource.

Folgende Tabelle gibt eine kurze Übersicht über das CAQ-Bruneck:

| Modul-name | Beschreibung |
|---|---|
| SPC83 | MDE Client, Statistische Prozesskontrolle und Regelung, Kontrolle der Prozessparameter, Automatische Gewichts-regelung T4, Reporting über Prozessverlauf |
| NPP | Prüfplanung |

| Modul-name | Beschreibung |
|---|---|
| FK | Behandlung interner Abweichungen |

**Tabelle 3: CAQ Bruneck Software Module**

Ein CAQ System wird allgemein in folgende Module gegliedert. Die folgende Tabelle gibt eine Übersicht über die verschiedenen Module die derzeit in Bruneck im Einsatz sind; inklusive der Information welche durch das CAQ Bruneck abgedeckt werden und wo Standardsoftware verwendet wird.

| Modul | Lieferant | Produkt / Modulname |
|---|---|---|
| Prüfplanung | CAQ Bruneck | NPP |
| Wareneingangsprüfung | CAQ Bruneck | SPC93 |
| Warenausgangsprüfung | CAQ Bruneck | SPC83 |
| Fertigungsprüfung - SPC | CAQ Bruneck | SPC83 |
| Prüfmittelmanagement | Böhme & Weihs Systemtechnik | CASQ-it 9000, PMW |
| Prüfmittelfähigkeit - MSA | Q-DAS GmbH & Co. KG | qs-STAT |
| Reklamationsmanage-ment (extern) | Plexus System LLC | Plexus |
| Reklamationsmanage-ment (intern) | CAQ Bruneck | FK Abweichungskarten |
| FMEA | APIS IT GmbH | APIS IQ-FMEA |
| APQP | Böhme & Weihs Systemtechnik | CASQ-it 9000 APQP |
| Controlplan | CAQ Bruneck/MS Word | SPC83 |
| Erstmusterprüfberichte | iqs Software GmbH | EMPB |

**Tabelle 4: CAQ System Bruneck Software Lieferanten**

## 4.9 Verbesserungspotential und Maßnahmen

Warum das System nun erneuert bzw. ersetzt werden soll, hat viele Gründe. Zum einen jene die, die Software betreffen, zum anderen aber auch Hardware bezogene Probleme.

Einer der wesentlichsten Gründe ist sicher, dass sich die Anforderungen an das System laufend erhöht haben, aber viele Änderungen nicht mehr wirtschaftlich implementiert werden können. Das liegt vor allem daran, dass die Software in die Jahre gekommen ist und der Umstieg auf eine moderne Entwicklungsumgebung die Tatsache mit sich bringt, dass der Programmcode neu geschrieben werden muss, da nur sehr wenige Routinen wiederverwendet werden können.

Weiters ist in Einzelfällen die Datenintegrität nicht mehr gewährleistet oder es gehen Daten verloren, da Dateien durch den gleichzeitigen Zugriff zerstört werden und diese dann mühsam über eine Sicherungskopie wiederhergestellt werden müssen.

Auch ein wesentlicher Punkt ist die Tatsache, dass die Auswertungen sehr schwer möglich sind, da die Daten in binärer Form abgelegt wurden und erst durch ein Programm in ein anderes Format konvertiert werden können. Dieser Vorgang ist sehr zeitaufwändig, da die Dateien z. T. mit verschiedenen Programmversionen abgelegt wurden und so auch verschiedene Konvertierungsprogramme geschaffen werden müssen.

Diese Tatsache ist auch dann relevant, wenn Schnittstellen mit anderen Programmen und Systemen geschaffen werden müssen.

Die Hardware, auf die das System abgestimmt ist, ist z. T. nicht mehr erhältlich, es sind z. B. noch ISA-Karten im Einsatz. Spezielle Mainboards dafür sind mittlerweile außerdem relativ teuer und kaum noch erhältlich.

All diese Punkte können nicht mehr punktuell gelöst werden, sondern am besten durch Migration auf ein neues System.

# 5 Technische Analyse des bestehenden Systems

Im Folgenden wird die Art und Weise beschrieben, wie die technische Analyse durchgeführt wurde. Der Projektverlauf wurde dabei in verschiedene Phasen unterteilt, die ebenfalls vorgestellt werden.

## 5.1 Projektverlauf und -ziel

Der Verlauf kann in drei Phasen gegliedert werden, in denen grundlegend zwei Szenarien verfolgt wurden, die Einführung des Produkts CASQ-IT 9000 der Firma Böhme und Weihs Systemtechnik Gmbh & Co KG oder Reengineering bzw. Renovation des bestehenden Systems.

In der ersten Phase wurde das bestehende System von Bruneck analysiert. Über mehrere Schritte wurden alle Anforderungen, die das System derzeit abdeckt, ermittelt. In der zweiten Phase wurden diese Anforderungen mit den Systemen von CASQ-IT 9000 verglichen und die Unterschiede herausgearbeitet. Dabei war sehr wohl klar, dass man das alte System dort nicht 1:1 wiederfinden würde, die Abläufe und Benutzerführung würden sich auf jeden Fall verändern.

Vielmehr wurde der Versuch unternommen, die Anforderungen mit dem System von B&W zur Deckung zu bringen und die Deltas herauszuarbeiten.

Dann, in der dritten Phase, wurden die gefundenen Deltas einzeln analysiert und auf ihre Bedeutung hin klassifiziert. Anschließend

wurde das Ergebnis an B&W übermittelt und auf Machbarkeit und Aufwand hin überprüft.

Das Ergebnis skizziert die zwei Szenarien in Bezug auf Aufwand, Deltas und Kosten. Beide Wege kann man nur zum Teil direkt miteinander vergleichen, da sie sehr unterschiedlich sind.

Das Dokument soll daher aufzeigen, welche Auswirkungen das Beschreiten des einen oder anderen Szenarios für das Unternehmen hat.

## 5.2 Reverse Engineering

Vereinfacht werden Software Systeme in drei Schritten erzeugt. Ausgehend von den Anforderungen (Spezifikation, Pflichten und Lastenheft) wird ein Design entworfen. Anschließend wird das System implementiert (programmiert). Diese Vorgehensweise wird auch Forward Engineering genannt. Geht man nun den Weg in die umgekehrte Richtung, dann erfolgt ein Reverse Engineering.

Das Reverse Engineering kann über eine oder mehrere Phasen erfolgen. Bei diesem Projekt war es nötig, bis zu den Anforderungen zurückzugehen. Es wurde in 3 Phasen eingeteilt:

1. Analyse des Quellcodes und des Systemumfelds
2. Design Modelle erstellen in UML Form
3. Requirements Erhebung

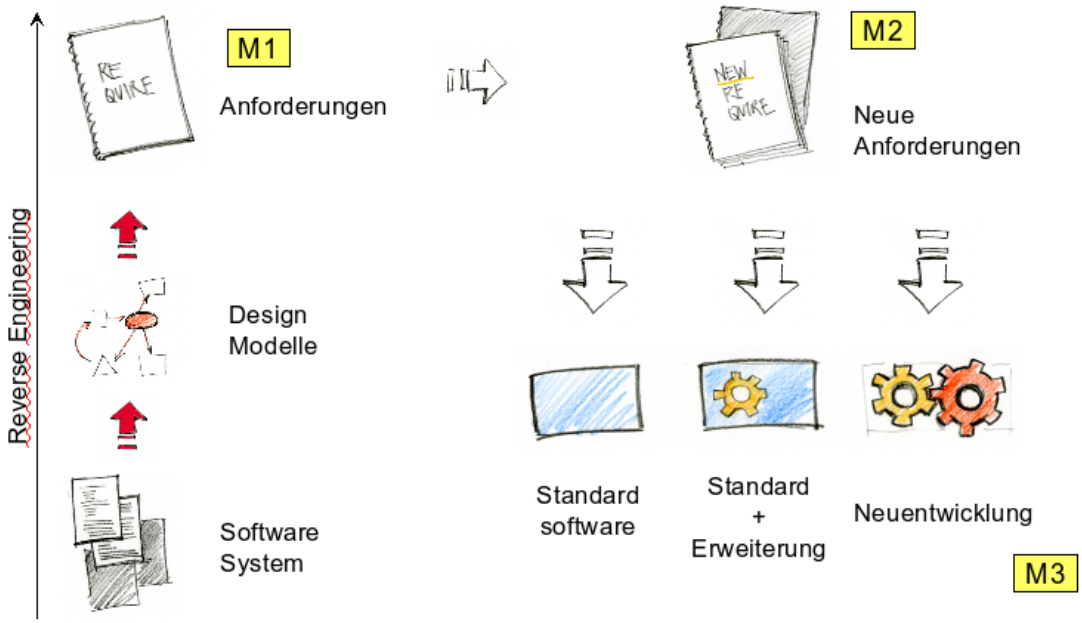


**Abbildung 23: Vorgehensweise**

Um die Anforderungen möglichst vollständig zu erfassen, wurde das System in einer ersten Phase analysiert. Die Analyse wurde auf drei Säulen aufgebaut:

- Analyse des bestehenden Programmcodes
- Benutzerbefragungen
- Apprenticing (lernen, das System zu benutzen)

In einer 2. Phase wurde das zugrundeliegende Design extrahiert und in UML Diagrammen abgebildet. Da diese Diagramme zu sehr auf die IT fokussiert sind, wurde in der dritten Phase ein Requirements Dokument verfasst, das das System und seine Abläufe sowie Constraints vorwiegend in Textform beschreibt.

Dann wurden die neuen Anforderungen hinzugefügt, aus denen sich ursprünglich 3 Szenarien ableiten sollten:

1. Einsatz reiner Standardsoftware
2. Einsatz von Standardsoftware und proprietären Erweiterungen
3. Reengineering bzw. Neuentwicklung des Systems

Der Punkt 2 wurde aber sehr schnell fallen gelassen, da die vorgegebene Standardsoftware nicht über entsprechende Schnittstellen verfügte und proprietäre Erweiterungen kaum durchführbar waren bzw. sich sehr kostenintensiv zu Buche geschlagen hätten.

Im Folgenden werden die einzelnen Säulen und Phasen genauer beschrieben:

### 5.2.1 Analyse des Quellcodes

Zu allererst ist anzumerken, dass es ein großer Vorteil dieses Projektes darin bestand, dass der Quellcode des Systems noch vorhanden war. Das System wurde in Turbo Pascal geschrieben. Jedoch war kein Versionsmanagement in Form einer Quellcodeverwaltung vorhanden. Die einzelnen Versionen der Dateien wurden durch unterschiedliche Endungen markiert bzw. wurden ganze Verzeichnisse auf diese Art und Weise gekennzeichnet. Der gesamte Quellcode war auf über 100 Dateien verteilt mit insgesamt ca. 100.000 LOC.

Da auf den ersten Blick ersichtlich war, dass auch eine große Anzahl an verwaisten Dateien, d. h. Dateien einer älteren Version oder nicht mehr benötigte Dateien, vorhanden waren, war es ratsam, ein Analyse Tool einzusetzen, das jenen Programmcode ermittelt, der bei einer Compilierung tatsächlich verwendet wird. Hierfür wurde Pascal Analyzer verwendet.

Das Ergebnis kann im Anhang im Detail betrachtet werden. Grundsätzlich besteht das System aus drei Hauptprogrammen sowie einem Programm zur Benutzerverwaltung.

- SPC83: Statistische Prozesskontrolle und Endkontrolle
- NPP40: Prüfplanung
- FK27: Bearbeitung von internen Abweichungen
- Editpass: Benutzerverwaltung

Alle zusammen verwenden zum Teil gemeinsame Programmbibliotheken. Zusätzlich sind auch Hilfsprogramme für Kontroll- und Testzwecke vorhanden, diese wurden aber nicht in die Analyse einbezogen. Nach kurzer Betrachung stellte sich heraus, dass sie für eine Reverseengeeniering irrelevant sind. In der Tabelle unten werden die 4 Programme und einige Parameter, die auf den Grad der Komplexität schließen lassen, aufgelistet.

| Programm | Files | Total Lines | LOC | Cmt Lines | Cmts/ Total | Cmts/ LOC | DP | DP/ LOC | Chrs/ LOC | CmtCh/ Chrs |
|---|---|---|---|---|---|---|---|---|---|---|
| SPC83 | 35 | 15242 | 12662 | 1389 | 0,09 | 0,11 | 4050 | 0,32 | 26 | 0,16 |
| NPP40 | 18 | 6814 | 5542 | 730 | 0,11 | 0,13 | 1571 | 0,28 | 25 | 0,19 |
| FK27 | 17 | 6498 | 5218 | 695 | 0,11 | 0,13 | 1337 | 0,26 | 25 | 0,18 |
| Editpass | 9 | 2906 | 2238 | 408 | 0,14 | 0,18 | 444 | 0,2 | 25 | 0,23 |
| Abbreviations | | | | | | | | | | |
| Chrs=Characters, Cmt=Comment, DP=Decision Points | | | | | | | | | | |
| LOC=Lines of Code, CmtCh=Comment Characters, Chrs=Character | | | | | | | | | | |

Tabelle 5: Programmcode Komplexitätsanalyse

Durch die Analyse der Reports (Anhang x) wird festgestellt, dass das Programm NPP40 fast vollständig im Programm SPC83 enthalten ist. Ebenso verhält es sich mit dem Programm Editpass, das zum Teil in den drei Hauptprogrammen verwendet wird. Somit bleiben von den 100.000 LOC nur noch ca. ein Viertel relevant, der Rest sind

Programmcode und Dateien, die ältere Versionen beinhalten oder nicht mehr relevant sind.

In der Tabelle oben fällt auf, dass sehr wenig Kommentare im Programmcode enthalten sind, das ist ein Typisches Merkmal von Legacy Sytemen. Genau wie die Tatsache, dass viele Programmzeilen nicht mehr verwendet werden.

Das Programm SPC83 ist das mit Abstand komplexeste Programm, diesem wurde auch die meiste Aufmerksamkeit gewidmet. Auch aufgrund der Tatsache, dass die Programme NPP40 und Editpass dort zum Teil verwendet wurden.

Bei der Betrachtung des Quellcodes fiel auf, dass User Interface Elemente, Programmcode und Datenhaltung nicht klar getrennt waren. Das erschwerte die Analyse, da vor allem dem UI keine Beachtung geschenkt werden sollte. Als nützlich hingegen hat hat sich die Definition der Datenstruktur erwiesen, aus der dann ein UML Diagramm erstellt werden konnte. Dies war der wertvollste Informationsgehalt, der aus dem Quellcode gewonnen werden konnte.

### 5.2.2 Apprenticing (in die Lehre gehen)

Das Apprenticing stellt eine weitere Säule dar, mit der Informationen gewonnen werden können. Apprenticing wird frei mit „in die Lehre gehen" übersetzt. Während diesem Prozess man, das System zu benutzen und findet die wichtigsten Abläufe und Prozesse heraus, die das System abbildet. Das war sehr hilfreich, da das System sehr komplex war und die Implementierung unübersichtlich. Doch schließlich gelang es, die wichtigsten Anwendungsfälle zu definieren und in einem UML Use Case Diagramm darzustellen.

Auffallend am System war das für damalige Verhältnisse ausgefeilte Sicherheitssystem. Es ermöglichte, BenutzerInnen Berechtigungen nur für bestimmte Funktionen zu erteilen. So kam es zwar vor, dass, wenn während kritischer Abläufe bestimmte Probleme auftraten, BenutzerInnen mit einer höheren Berechtigung, z. B. VorarbeiterInnen, diese Probleme freigeben mussten. Es hat sich aber gezeigt, dass die Software hier bereits in hohem Maße die Arbeit der MitarbeiterInnen unterstützte und sie zwang, bestimmte Abläufe einzuhalten.

Ein auffälliger Aspekt war weiters, dass das gesamte System nur über eine Tastatur bedienbar war. Bei der Beobachtung der MitarbeiteInnen fiel auf, dass sie Daten sehr schnell eingeben konnten, viel schneller, als es eine Bedienung mit einem Pointer (Maus) schaffen würde.

All diese Erkenntnisse wurden dokumentiert und, aufgrund ihrer Bedeutung für den reibungslosen Ablauf, kategorisiert. Einige von ihnen wurden dann als funktionale Anforderungen (Constraints) für das neue System aufgenommen.

Dazu gehörten z. B.:

- Die Benutzung des neues Systems erfolgt ausschließlich über die Tastatur. Auch ist es wünschenwert, dass man Tastaturkürzel (short keys) frei definieren kann, damit die bestehenden Kürzel übernommen werden können – mit dem Ziel, den MitarbeiterInnen den Einstieg in das neue System zu erleichtern.
- Eine Messstation darf maximal für 15 Minuten down sein, ansonsten kann ein reibungsloser Betrieb nicht mehr garantiert werden.

### 5.2.3 Benutzerbefragungen

Begleitend zur Analyse des Programmcodes und zum Apprenticing ergaben sich viele Fragen. Diese Fragen konnten dann an Ort und Stelle mit den entspechenden MitarbeiterInnen diskutiert werden. Dabei kamen verschiede Aspekte zum Vorschein.

Im Programmcode gab es verschiedene Zeilen, deren Bedeutung nicht zugeordnet werden konnte. Dabei wurde festgestellt, dass Dinge einprogrammiert wurden, die später nie verwendet wurden, da sie an anderen Dingen gescheitert waren.

Z. B. waren Messvorrichtungen im Programmcode vorgesehen, wurden aber nie realisiert, da es messtechnische Probleme gab, bzw. die Vorrichtungen nicht den gewünschten Vorteil brachten.

Auf der anderen Seite wurde festgestellt, dass Abläufe und Datenfelder im Programm ausgefüllt wurden, obwohl sie nicht mehr benötigt wurden.

Allgemein kann man sagen, dass durch die begleitende Maßnahme der Benutzerbefragungen, viele vermeidbare Anforderungen nicht den

Weg in das abschließende Requirements Dokument geschafft haben und dieses Dokument so nicht unnötig aufgebläht wurde.

### 5.2.4 Ergebnisse des Reverse Engineering

Die Ergebnisse dieser drei Säulen wurden in Form von UML [25]dokumentiert. Die ursprüngliche Systembeschreibung war nur noch zum Teil bzw. gar nicht vorhanden oder veraltet. UML als Dokumentationssprache zu verwenden, erscheint angesichts des prozeduralen Quellcodes auf den ersten Blick nicht sinnvoll. Aber die Dokumentation hatte das Ziel, das System auf einer höheren Ebene – der Design und Requirements-Ebene – zu dokumentieren, dabei war UML sehr gut geeignet. Zudem stellt es einen Standard da und die erstellten Diagramme können im Fall eines Forward Engineerings wiederverwendet werden.

Zusätzlich wurden einfache Textdokumente erstellt, um diese Dokumentation zu ergänzen. Dabei entstand auch eine Systembeschreibung, bzw. eine Art Benutzerhandbuch, das vorher nicht vorhanden war.

Das Wissen, wie das System zu benutzen ist, wurde intuitiv erarbeitet oder einfach von Mitarbeitern weitergegeben. Diese Methode der mündlichen Überlieferung ist weit verbreitet aber eines Technologiekonzerns nicht würdig.

Im Folgenden wird eine Auswahl an UML Diagrammen in Form von Kurzbeschreibungen vorgestellt.

**Kernprozess**

Der Prozess beginnt in der Fertigung, hier wird erstmals ein Prüfplan angelegt. Voraussetzung dafür ist, dass das Produkt bereits im

System angelegt wurde, d. h. die Produktdaten wurden aus dem BaaN übernommen.

Die Fertigung legt alle Prozesse an, sie kann aber nur produktspezifische Merkmaldaten (z. B. Nennmaß, Toleranz, Einheit etc) eingeben. Der Prüfplan wird gesperrt und dann von der Qualität weiterbearbeitet. Dabei werden die qualitätsspezifischen Merkmaldaten (Eingriffsgrenze, Cpk Wert, Prüfmittel, Stichprobengröße etc.) ergänzt und der Prüfplan freigegeben. Welche Felder genau von wem bearbeitet werden können, geht aus dem Requirement Prüfplan hervor.

**Maschinenstatus für MDE Meldevorgang**

Um die MDE durchzuführen, werden verschiedene Meldungen abgesetzt, eine Meldung besteht dabei aus folgenden Einträgen:

- Datum und Zeit
- Qualitätsmerkmal, wird nur angegeben, wenn die Meldung ein Merkmal betrifft.
- Aktuell produzierte Stückzahl
- Bemerkung

Eine MDE Meldung signalisiert auch, dass sich eine Maschine von dem Zeitpunkt an in einem bestimmten Zustand befindet.

Wird das Rüsten bzw. das Fertigen unterbrochen, kann eine Bemerkung hinzugefügt werden. Die Bemerkungen sind dabei vorgegeben, sie können in einer Datei frei definiert werden. Auch beliebige Bemerkungen sind möglich. Jede Bemerkung erfordert eine entsprechende Berechtigungsstufe der BenutzerInnen..

**Usecase Prüfplanung**

Bei der Prüfplanung wird festgelegt, wie die Qualitätsmerkmale während des Fertigungsprozesses gesichert und in welchen Intervallen sie wie oft gemessen werden. Den Plan selbst kann man sich wie ein normales Dokument vorstellen, das mit einer bestimmten Benutzerberechtigung gesperrt und freigegeben werden kann. Für dieses Dokument wird automatisch eine Übersicht über alle Änderungen erstellt, die vorgenommen wurden.

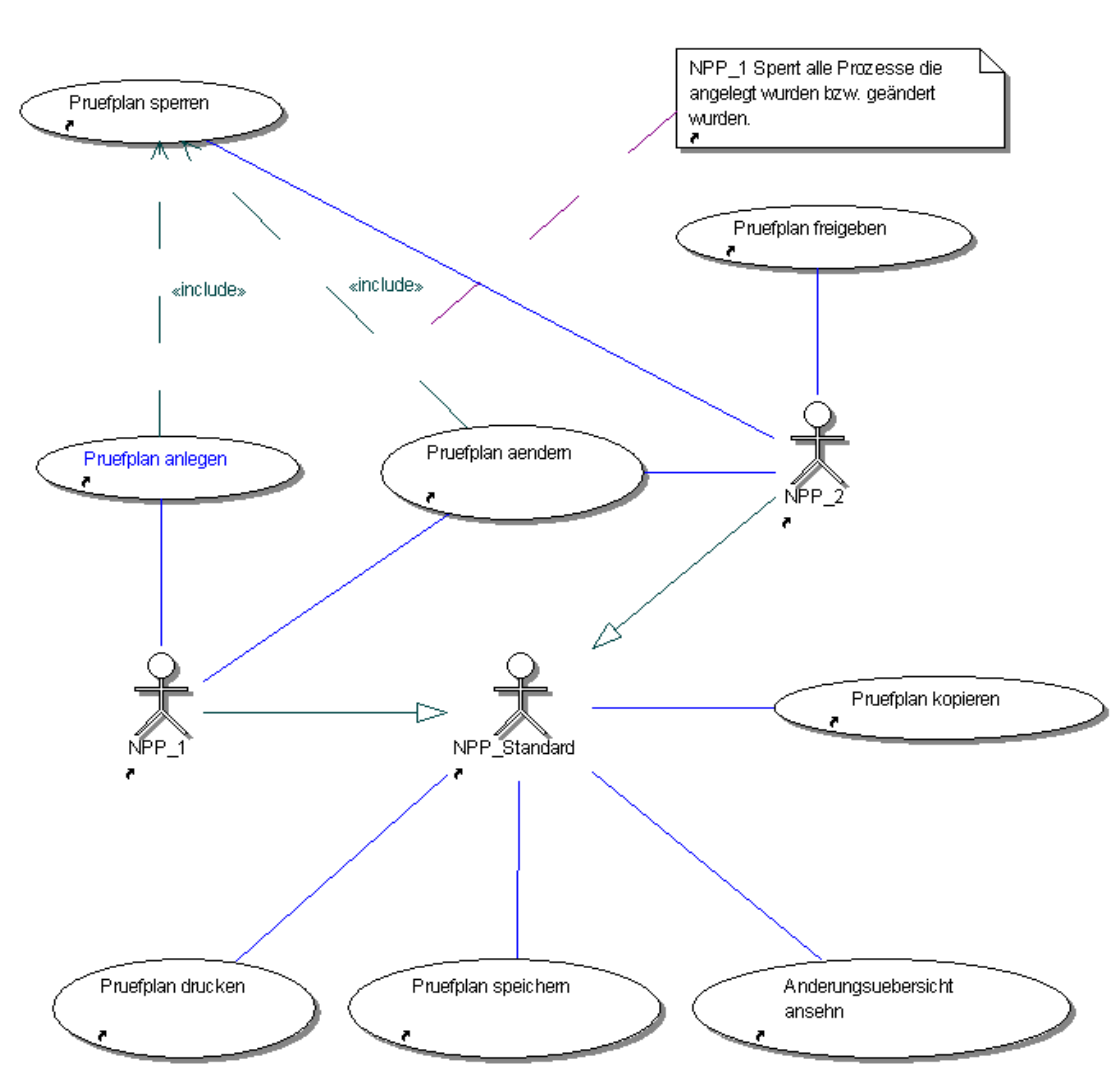


**Abbildung 24: Use Case Diagramm Prüfplanung**

**Use Case SPC**

Die Durchführung der SPC ist ein sehr komplexer Vorgang. Grundsätzlich haben viele Anwendungsfälle Pre- und Postconditions. Zentraler Anwendungsfall ist das Erfassen der Messwerte, in dem im Prüfplan vorgegebenen Intervall, sowie deren Auswertung i. a. die Ansicht der verschienenen Fehlersammel- und Regelkarten.

Hinzu kommt das Absetzen von MDE Meldungen, die laufend in das System eingegeben werden können.

Die SPC beginnt mit dem Öffnen eines Fertigungsauftrags. Hier werden z. T. auch Prozessparameter eingegeben, doch dies ist ein eigener Anwendungsfall, der nicht immer durchgeführt werden muss. Außerdem werden bestimmte Parameter auch validiert, z. B. wird bei einem Chargenwechsel die neue Chargennummer eingegeben und diese dann in einer eigenen Datenbank überprüft, ob sie auch das richtige Material für diesen Auftrag enthält.

Hinzu kommen Anwendungsfälle wie das Wechseln der Schicht, das Drucken der Dokumentation etc.

Ein wichtiger Punkt ist auch der Bereich, der als T4 gekennzeichnet ist. Es handelt sich hier um eine spezielle Produktionsmaschine, mit der das Gewicht der Teile bzw. die Dichte automatisch geregelt wird. Der Aktor hat die Möglichkeit, diese zur Regelung zu aktivieren bzw. zu deaktivieren. Ist diese Regelung aktiv, wird das Programm selbst zum Aktor, der in definierten Intervallen die Messung der Dichte vornimmt und dann, im Falle einer Abweichnung, entscheidet, ob die Maschine nachgeregelt oder, sollte die automatische Nachregelung nicht mehr möglich sein, die Produktion unterbrochen werden muss.

Das gesamte System ist auch mit Berechtigungen versehen, jeder Aktor darf bestimmte Anwendungsfälle durchführen, bzw. Teile

davon. Z.B. ist jede Entscheidung nach einer Abweichung mit einem Berechtigungslevel versehen.

Ist ein Auftrag beendet, wird er geschlossen, dabei müssen alle offenen Entscheidungen gefällt werden, sonst ist ein Schließen nicht möglich.

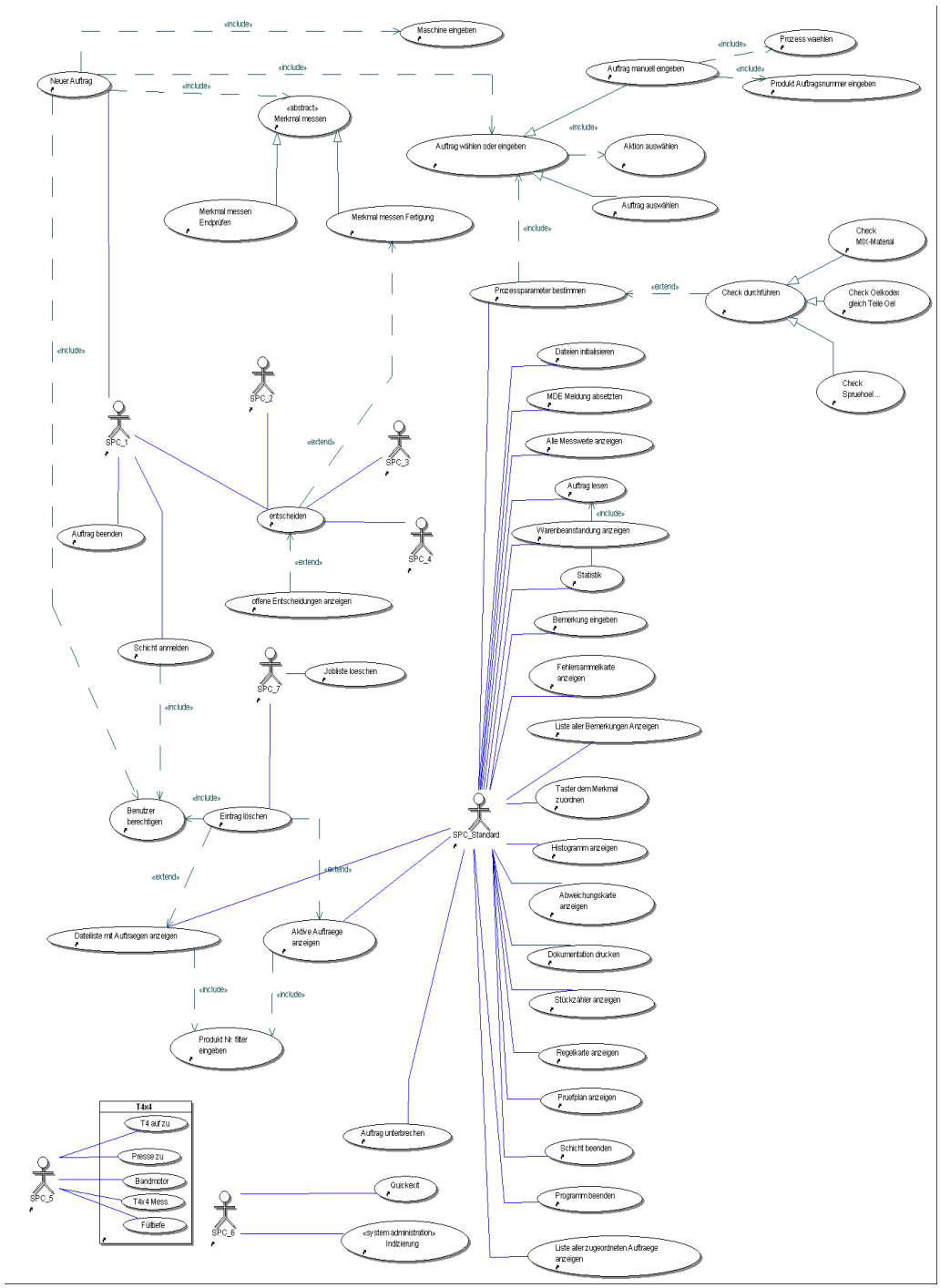


**Abbildung 25: Use Case Diagramm SPC**

### Aktivity Diagramm Prüfplan eingeben

Ein Prüfplan besteht aus einer hierarchischen Ordnung von

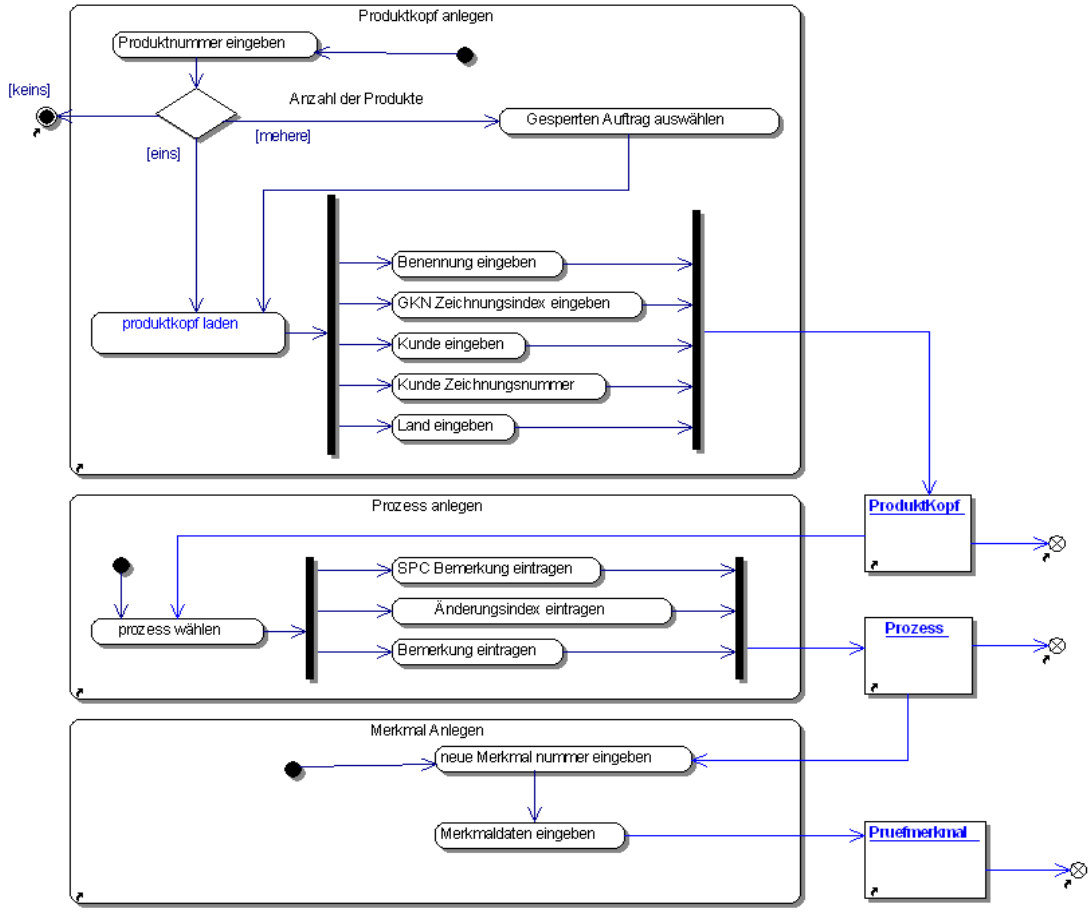


**Abbildung 26: Aktivity Prüfplan anlegen**

Produktkopf, einem oder mehreren Prozessen sowie und den dazugehörigen Qualitätsmerkmalen. Jeder Prüfplan wird einem Produkt zugeordnet. In jeder Hierachieebene müssen diverse Datenfelder ausgefüllt werden, sofern die entsprechende Berechtigung an die BenutzerInnen erteilt wurde. Am Ende der Aktivitäten entsteht der fertige Prüfplan.

## 5.3 Deckung der Anforderungen mit CASQ-IT 9000

Die erhobenen Anforderungen aus Phase 1 wurden als Grundlage für diese Phase verwendet. Einige Module von CASQ-IT 9000 sind im Werk Bad Brückenau im Einsatz. Aus diesem Grund wurde dort ein zweitägiger Workshop organisiert, um die Delta Analyse der beiden Systeme durchzuführen. Hierzu waren Experten aus Bad Brückenau, Bruneck und von B&W anwesend.

Dass die Funktionen von CAQ Bruneck nicht im System von B&W wiedergefunden werden können, wurde eingangs erwähnt und war bereits im Vorfeld klar. Somit haben wir uns auf folgende Vorgehensweise geeinigt:

Wir gehen die Abläufe, wie sie das System von B&W vorsieht, anhand von Beispielen durch. Dabei wird ersichtlich, welche Teile der Anforderungen abgebildet sind. Dass die Abläufe und Benutzerführung zum Teil grundverschieden sind, davon musste man ohnehin ausgehen – diese Sicht war daher nicht relevant.

Ziel war es einerseits, mögliche Probleme bei der Einführung in diesem frühen Stadium zu erkennen und andererseits, all jene Anforderungen, die das System nicht oder nur zum Teil abdeckt, zu finden und zu dokumentieren.

### 5.3.1 Delta Analyse

Eine Reihe von Punkten konnte mit dem System CASQ-IT 9000, so wie es in BBR in Betrieb ist, nicht abgedeckt werden. Daher wurde in einer gemeinsamen Diskussion entschieden, auf welche Punkte verzichtet werden können, bzw. wo es keinen Sinn macht, sie mit

dem System CASQ-IT 9000 abzudecken. Die anderen Punkte wurden durch drei Prioritätsstufen klassifiziert:

- Die höchste bedeutet, dass diese Anforderung unbedingt im System enthalten sein muss.
- Die mittlere bedeutet, dass die Effizienz der Abläufe beeinträchtigt wird.
- Die niedrigste bedeutet, dass eine Anforderung wünschenswert, aber nicht zwingend erforderlich ist.

Am Ende wurden diese Punkte an B&W übermittelt, wo Machbarkeit, Aufwand und Kosten für ein entsprechendes Customizing analysiert wurden. Einige Bereiche, die das System CAQ Bruneck abdeckt, sind nicht im Lieferumfang von B&W enthalten. Diese wurden nicht angefragt. Insbesondere handelt es sich um den MDE Client, ein Parameter Kontrollsystem, gemeinsames Reporting aller Prozesse und SPC Daten mit schneller Rückverfolgung.

### 5.3.2 Change Prozess

Durch die Einführung einer neuen Software wird ein Change Prozess eingeleitet. Ein solcher Change Prozess folgt stets einem ähnlichen

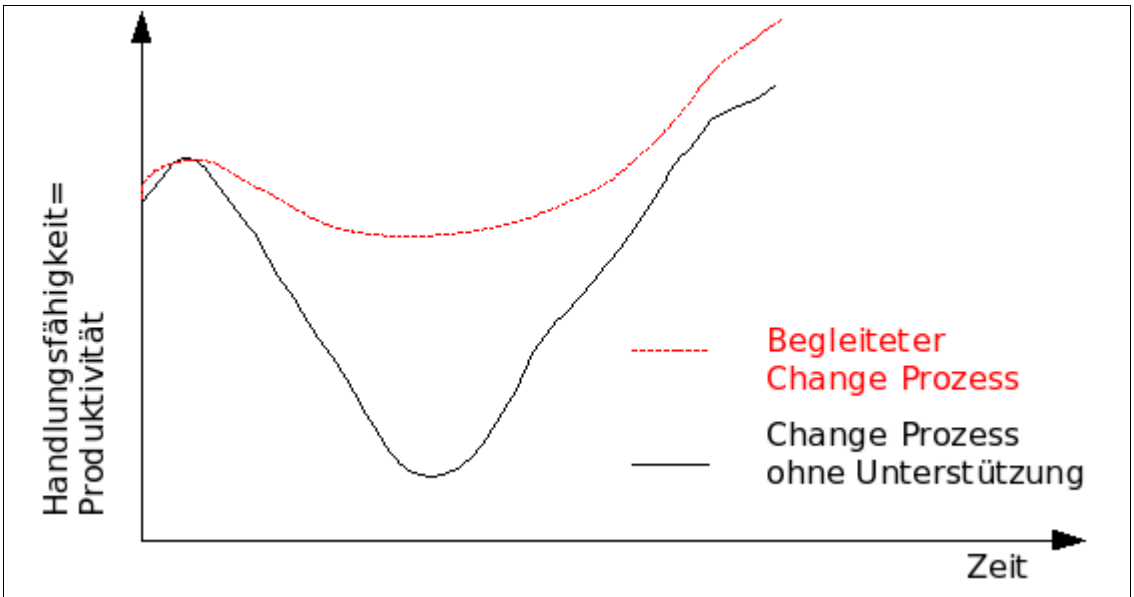


**Abbildung 27: Changeprozess**

Muster. In diesem konkreten Fall sind etwa 400 MitarbeiterInnen betroffen und die Auswirkungen dürfen nicht unterschätzt werden. Eine detaillierte Planung, wie der Umstieg erfolgen soll und eine begleitende Unterstützung aller Beteiligten ist unbedingt notwendig, um einer Minderung der Produktivität entgegen zu wirken.

Durch die Einführung eines neuen Systems werden Abläufe im Unternehmen verändert. Anfänglich entsteht eine kurze Begeisterung für das Neue. Doch meistens erzielen Veränderungsprozesse erst nach einiger Zeit die erwarteten positiven Ergebnisse. Vorher durchwandern alle Beteiligten eine sogenannte „Konfusionsphase", in der der Umgang mit dem neuen System erarbeitet werden muss. In dieser Zeit kann man sich auf eine Verschlechterung der Kennzahlen einstellen.

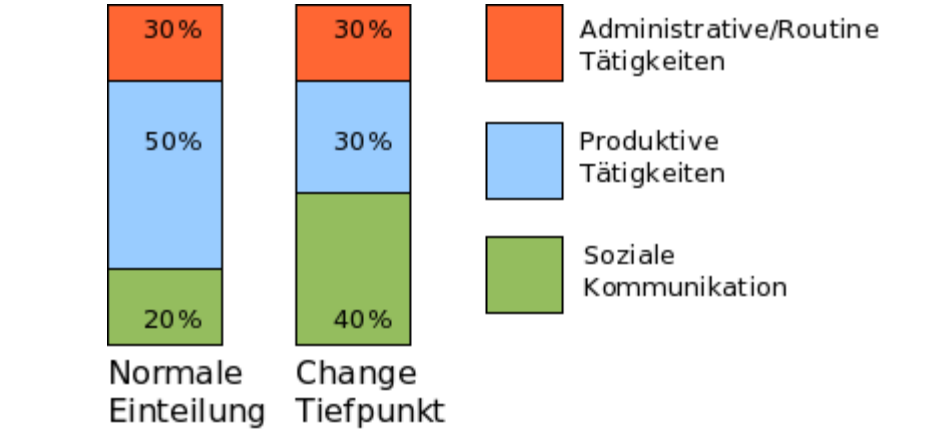


**Abbildung 28: Tätigkeiten im Unternehmen**

In einer Panik-Falle steckt der Prozess in dieser Situation dann, wenn einige Manager panikartig mit mehr Druck und verstärkter Kontrolle auf diese vorübergehende Verschlechterung reagieren. Im Extremfall gehen Mitarbeiter dann sogar auf Distanz zum laufenden Change Prozess und entziehen dem Projekt dadurch den Rückhalt. Die Produktivität ist an einem Tiefpunkt angelangt, es besteht die Gefahr,

dass das Projekt scheitert. In dieser Zeit wird viel kommuniziert, um der Unzufriedenheit Luft zu verschaffen.

Um solchen Entwicklungen entgegen zu wirken, ist es wichtig, den Change Prozess im Idealfall mit einem Change Team zu begleiten.

Beide Szenarien läuten im konkreten Fall einen Change Prozess ein. Bei einer Einführung von CASQ-it 9000 werden die Veränderungen an den Prozessen umfangreicher sein. Somit wird auch das Risiko, dass der Change Prozess nicht erfolgreich durchgeführt werden kann, größer sein.

# 6 Evaluierung möglicher Lösungsszenarien

Nach dem Ansatz, eine Lösung der Problemstellung durch Skizzieren verschiedener Szenarien zu finden, werden hier zwei Möglichkeiten ausführlich dargestellt. Ursprünglich waren drei Szenarien geplant, das Szenario das nicht weiter verfolgt wurde, war eine Mischung aus den beiden anderen, wurde aber verworfen, da es keine ausreichenden Schnittstellen an der Standardsoftware gab. Im Folgenden werden die beiden Szenarien *Einsatz einer Standardsoftware* und *Reengineering* näher beschrieben:

## 6.1 Szenario Standardsoftware

Dieses Szenario beschreibt den möglichen Einsatz einer Standardsoftware, ohne dabei zu berücksichtigen, dass diese durch Schnittstellen erweitert werden könnte. Der Einsatz soll rein durch Customizing und marginale Anpasssungen erfolgen.

### 6.1.1 Allgemeine Szenariobeschreibung

Diese Lösung besteht aus verschiedenen Modulen, die zum größten Teil untereinander kompatibel sind. Sie bilden die Basisforderungen der ISO TS 16949 ab. Folgende Module werden in der Basisversion von CASQ-IT 9000 bereitgestellt und sind bereits durch einen Rahmenvertrag für die gesamte Sintermetals abgedeckt.

- CPL Control Plan

- EMPB Erstmusterprüfbericht[1]
- ZP Zwischenprüfung
- SPC/QDE Qualitätsdatenerfassung
- RUF Reklamationsverfolgung, Fehlerkostenanalyse
- PMW Prüfmittelwesen[2]

Die Umstellung wird mit einem erheblichen Schulungsaufwand verbunden sein, da sich die Benutzerführung, die Oberfläche aber auch die Prozesse (Abläufe), die von der Software abgebildet werden, ändern.

Es wird ein umfangreicher Change Prozess eingeleitet. Die Abläufe in Bruneck wurden über die Jahre für die Anforderungen optimiert, die an die CAQ gestellt wurden. Einige dieser optimierten Verfahren können bei der Umstellung auf CASQ-IT 9000 nicht mehr durchgeführt werden, da sie von der Software in dieser Form nicht unterstützt werden (z. B. Optimierte Rundlaufmessung). Daher müssen bestehende Abläufe umgestellt und neu optimiert werden.

Generell wird eine Umstellung mindestens ein Jahr in Anspruch nehmen. In dieser Zeit werden das CAQ System Bruneck und das neue System mit CASQ-IT 9000 parallel laufen. Das bedeutet, dass beide Systeme gewartet werden müssen. Die bestehenden Daten können nur zum Teil nach B&W und nur einmalig übernommen werden. In BBR mussten alle Prüfpläne händisch überarbeitet werden. Es ist hier dringend anzuraten, einen genauen Projektplan für die Einführung zu erstellen, da ohnehin eine gewisse Unruhe in Fertigung und Qualität zu erwarten ist.

Zeitgleich mit der Einführung des Standardsystems CASQ-IT 9000 müssen auch die Zusatzsysteme bereitstehen, um einen

[1]Kann in Bruneck nur bedingt verwendet werden, da der Kunde Bosch ein anderes Programm verlangt.
[2]Wurde 2006 eingeführt.

reibungslosen Betrieb aufrecht zu erhalten. Daher ist es wichtig, rechtzeitig mit der Spezifikation und Entwicklung dieser Systeme zu beginnen.

### 6.1.2 Deltas unbedingt nötige Anforderungen

Die Anforderungen, die das System von B&W nicht abdeckt, wurden ermittelt und kategorisiert. Auf einen Teil musste von vorne herein verzichtet werden, die anderen wurden nach ihrer Wichtigkeit in low, medium und high klassifiziert. Die folgende Liste gibt eine Übersicht über die **unbedingt benötigten Anforderungen**, um eine effiziente Arbeitsweise aufrecht zu erhalten.

Grundsätzlich gilt, dass bei diesem Szenario die Prozesse und Strukturen stärker verändert werden. Deshalb sollte auch dem Change Gedanken verstärkt Rechnung getragen werden.

**Messauftrag zurücksetzen**

Beim Durchführen eines Produktionsauftrags an der SPC Station dürfen andere Produktionsaufträge nicht einfließen. Jeder Auftrag steht für sich, die Messungen und daraus resultierenden Prozesskennzahlen, insbesondere der Cpk Wert, werden immer nur für einen Auftrag an der SPC Station berechnet, d. h. Messergebnisse werden an der SPC Station immer pro Artikel, Maschine und Auftrag betrachtet.

**B&W Status:**

Im Customizing vom Werk Lichfield enthalten.

**Prüfintervall pro Stück**

Prüfintervalle müssen auch pro Stück angegeben werden können. Dabei ist die Koppelung an einen Stückzähler erforderlich.

**B&W Status:**

Prinzipiell möglich über den Umweg einer Messvorrichtung.

Erforderliche Lizenzen und Projektarbeiten:

- Upgrade auf CASQ-it SPC/QDE Expert erforderlich
- Zukauf von CASQ-it MESS Messwertübernahme in die Datenbank
- Projektarbeiten zum Abstimmen der Vorrichtung, keine Angabe des Aufwands seitens B&W!

**Bei Maschinenstillstand Zeitbalken anhalten**

Wenn eine Maschine stillsteht, dann muss die Zeit, welche den nächsten Messzeitpunkt festlegt (signalisiert durch den Fortschrittsbalken) angehalten werden.

**B&W Status:**

Realisierbar durch Einführung des CASQ-it Fertigung/MDE Moduls. Hier gilt es anzumerken, dass bereits eine MDE Software in Verwendung ist.

- CASQ-it 9000 Fertigung / MDE
- Projektarbeiten zur Abstimmung und Anpassung, keine Angabe des Aufwands seitens B&W!

**Merkmalnummerierung**

Das Nummerierungssystem für Merkmalnummern muss beibehalten werden können. Jede Merkmalnummer besteht aus einer Nummer von 1 bis 999 mit maximal 2 Nachkommastellen. Das Kommazeichen wird als Punkt dargestellt.

Beispiele: 8 17.1 117.23

**B&W Status:**

Nicht möglich. Nur ganze Zahlen.

### Messmittelkanäle und Prüfsequenzen

Sowohl Messmittelkanäle als auch die Prüfsequenz müssen am Prüfplatz frei definiert werden können. Der Aufbau am Arbeitsplatz ist variabel und kann vorher nicht festgelegt werden.

**B&W Status:**

Die freie Festlegung der Messmittelkanäle ist an der Messstation bei Tastern möglich, so wie es in der SPC-Version für GKN Bad Brückenau eingerichtet wurde.

Das wahlweise Prüfen fälliger Merkmale kann durch die Merkmalvorauswahl realisiert werden. Das bedeutet einen erheblichen Mehraufwand bei jeder Messung.

### Änderungshistorie

Die Änderungshistorie muss aktiviert werden, jede Veränderung am Prüfplan muss mit Datum und Uhrzeit nachvollziehbar sein, da es vorkommen kann, dass ein Prüfplan mehr als einmal am Tag geändert wird. z. B. bei Versuchen.

**B&W Status:**

Möglich in der SPC/QDE Expert Version

### Versionsmanagement

Prüfpläne werden zum Teil sehr häufig geändert (mehr als einmal pro Tag). Wird im CAQ System Bruneck ein neuer Produktionsauftrag gestartet, wird dieser mit der aktuell gültigen Fassung des Prüfplans durchgeführt und bei Auftragsabschluss werden beide zusammen abgelegt. Der gespeicherte Prüfplan bleibt unverändert, auch wenn am Prüfplan nachfolgend Änderungen vorgenommen werden. Werden zu einem späteren Zeitpunkt die Daten des Produktionsauftrags geöffnet, dann hat man unmittelbar jene Version des Prüfplans zur Verfügung, mit der der Auftrag gefertigt wurde. Ein äquivalentes System muss zur Verfügung gestellt werden.

**B&W Status:**

Möglich durch Einführen des

- CASQ-it 9000 Moduls Versionsmanagement

### Kennzeichnung von wichtigen Merkmalen

Wichtige Merkmale (Neben-, Haupt- und kritische Fehler) müssen sowohl im Controlplan, als auch an der SPC Station gekennzeichnet werden.

**B&W Status:**

Wird kostenfrei zur Verfügung gestellt zur Wahrung der Normkonfomität.

### Urwertkarten an der SPC Station

An den SPC Stationen müssen neben den Xquer/s Karten auch Urwertkarten verfügbar sein. In Bruneck werden alle Variablen Merkmale mit SPC Regelkarten erfasst. Handelt es sich um kein klassisch regelbares Maß, wird anstelle der Xquer/s Kate eine Urwert Karte verwendet.

**B&W Status:**

Nicht möglich.

### Freie Bemerkungen

SPC Meldungen müssen zusätzlich zu den Auswahllisten (Katalogfelder) auch frei als Text eingegeben werden können.

**B&W Status:**

Derzeit nicht möglich, könnte in einer nächsten Release aufgenommen werden.

### Berechtigungssystem für Freigaben und Abweichungen

Im aktuellen System von GKN können für die unterschiedlichen Freigaben (Ausschuss, Nacharbeit, Sonderfreigabe) unterschiedliche Berechtigungsstufen vergeben werden. Reichen die Benutzerrechte für eine

Freigabe – z.B. Sonderfreigabe – nicht aus, verlangt das System, dass sich BenutzerInnen mit einer höheren Berechtigung anmelden und die Freigabe erteilen.

**B&W Status:**

Realisierbar durch Einführung und Upgrade von Modulen.

- Upgrade CASQ-it 9000 RUF Standard auf Expert
- CASQ-it 9000 QM Project.

### Automatische Berichterstellung – Kundendokumentation

Am Ende eines Produktionsauftrags soll automatisch ein Messbericht, abhängig vom Teil, Auftrag, Kunden in der Sprache der KundInnen (zumindest deutsch oder englisch) generiert werden, welcher durch eine berechtigte Person freigegeben wird.

**B&W Status:**

Realisierbar durch Customizing des Systems mit Projekarbeiten.

### Dokumentationsvorlagen

Für die Berichterstellung Kundendokumentation müssen verschiedene Vorlagen frei definiert werden können.

**B&W Status:**

Realisierbar durch Einführung einer Schnittstelle und Projektarbeit.

- CASQ-it Interface zu MS Automation Server
- Pro Schnittstelle 1-3 Tage Projektarbeit

### Messtaster bei End- und Zwischenprüfung

Bei dem Modul für End- und Zwischenprüfung muss man Messtaster (Induktivtaster) einsetzen können.

**B&W Status:**

Nicht möglich.

**Cpk Wert Abweichung**

Eine Cpk Abweichung muss wie eine Merkmalsverletzung behandelt werden.

**B&W Status:**

Realisierbar durch Customizing des Systems mit Projekarbeiten im Ausmaß von 3 Tagen.

**Hintergrundmessung und T4**

Es muss die Möglichkeit bestehen, Messwerte automatisch einzugeben und dann freizugeben. Z. B. 3D Messgerät. Für den Spezialfall der automatischen Gewichtsmessung an den T4 Maschinen, muss das System eine Messung starten. Dabei müssen alle Informationen über eine Schnittstelle zu einem Merkmal (Eingriffsgrenze, Toleranz, Nennmass etc.) übertragen werden. Danach muss das Messergebnis wieder über eine Schnittstelle eingelesen werden.

**B&W Status:**

Schnittstelle möglich, dafür erforderliche Lizenzen und Projektarbeiten:

- Zukauf von CASQ-it MESS Messwertübernahme in die Datenbank
- Projektarbeiten zum Abstimmen der Vorrichtung, keine Angabe des Aufwands seitens B&W!

### 6.1.3 Auswirkungen nicht gedeckter Anforderungen bei Einführung von CASQ-IT 9000

Im folgenden werden die Auswirkungen der nicht gedeckten Anforderungen erläutert, die bei einer Einführung von CASQ-IT 9000 im Werk Bruneck zu erwarten sind.

**Merkmalnummerierung**

Derzeit werden Merkmale mit einem für das Unternehmen zugeschnittenen System gekennzeichnet. Dadurch kann man unmittelbar aus der Nummer schließen, um welche Art von Merkmal (Länge, Durchmesser, Gewicht etc. ) es sich handelt. Merkmalnummern können dabei nicht nur ganzzahlig, sondern auch in der Form 1.1, 1.32 auftreten. Dabei gibt der erste Teil vor dem Punkt immer Auskunft über die Art.

Bei B&W werden die Merkmalnummern rein als eindeutiger ganzzahliger Schlüssel verwendet und haben keine Bedeutung.

**Auswirkungen:**

Alle GKN Zeichnungen und Dokumentationen der lebenden Produkte, ca. 3.000 Stück, müssen dahingehend überarbeitet werden. Sinnvoll ist wahrscheinlich, dass man sich generell von diesem System der Merkmalnummerierung (mit Punkt) trennt.

**Urwertkarten an der SPC Station**

Derzeit werden Urwertkarten an der SPC Station eingesetzt, das SPC Modul von CASQ-IT 9000 bietet nur eine Xquer/s Karte an der Messstation. Urwertkarten werden vor allem dann eingesetzt, wenn die Regelung eines variablen Merkmals nicht möglich ist. Das ist z. B. bei der Wärmebehandlung und der Messung der Härte nach Vickers der Fall. Sie besitzt eine starke Streuung und der Prozess gilt als nicht regelbar.

Haben die Messwerte eine hohe Streuung, so tritt bei der Xquer/s Karte eine Verletzung der Streuung auf und somit wird eine Abweichung gemeldet. Diese Abweichung ist aber nicht von Bedeutung, denn es reicht aus, dass die Härte sich innerhalb der vorgegebenen Grenzwerte befindet.

Schaltet man die Streuung aus und zeigt nur den Mittelwert an Xquer Karte, dann werden einzelne Abweichungen bei einer Stichprobe nicht mehr erfasst, da sie bei der Bildung des Mittelwerts verloren gehen.

Zudem reagiert die Xquer/s Karte verzögert, das heißt, es kann sein, dass eine Abweichung erst bemerkt wird, wenn der Produktionsauftrag bereits abgeschlossen ist.

**Auswirkungen:**

Der Einsatz von Xquer/s Karten generiert möglicherweise Abweichungen der Streuung, die unerheblich sind, aber als Abweichung beim Kunden aufscheinen.

Bildet man nur den Mittelwert, ist der Vorgang nicht hinreichend genau und es besteht die Gefahr, dass fehlerhafte Teile zum Kunden gelangen.

### Freie Bemerkungen

Derzeit können jederzeit freie Bemerkungen in Textform eintragen werden, was mit B&W nicht möglich ist.

**Auswirkungen:**

Künftig müssen alle Bemerkungen in vordefinierten Katalogen erfasst werden. Nicht katalogisierte Abweichungen können nicht erfasst werden.

Zusatzinformationen, z. B. besondere Vorkommnisse, Probleme können nicht erfasst werden und gehen somit verloren.

### Messtaster für End- und Zwischenprüfung

Bei B&W sind die Vorgänge der End- und Zwischenprüfung, sowie der SPC-Prüfungen auf die Module für die SPC und ZP aufgeteilt. Da es sich bei dem Modul ZP um einen Zukauf handelt, der noch nicht vollständig integriert ist, ist hier nicht der volle Funktionsumfang zu erwarten, der bei den SPC Stationen vorhanden ist. B&W konnte

keinen Termin nennen, wann die vollständige Integration vollzogen ist. Derzeit ist es nicht möglich, Messtaster an den Stationen für Zwischen- und Endprüfung zu verwenden.

**Auswirkungen:**

Messungen mit Tastern (z. B. Rundlauf) müssen künftig für Zwischen- und Endprüfung manuell durchgeführt werden. Das birgt ein erhöhtes Risiko einer falschen Dateneingabe.

**Rundlaufmessung**

Die Rundlaufmessung wird derzeit nach einem über die Jahre angepassten Verfahren durchgeführt. B&W unterstützt diese Art der Messung nicht. Sie muss auf ein Verfahren von B&W umgestellt werden. Es erfolgt über Fußtaster, mit denen der Start- und Endzeitpunkt bekanntgegeben wird, die Messung wird von einer Ibrit Messbox durchgeführt. Grundsätzlich ist das System nicht so effizient wie das Verfahren beim CAQ System Bruneck.

Weiters ist zu klären, wie eine Rundlaufmessung bei der Zwischen- und Endprüfung durchgeführt werden soll.

**Messmittelkanäle und Prüfsequenzen**

Derzeit kann ein(e) MitarbeiterIn frei entscheiden, ob er/sie eine Messung zu einem früheren Zeitpunkt durchführt und in welcher Reihenfolge Merkmale gemessen werden. Mit CASQ-IT 9000 ist die Reihenfolge in einer Prüfsequenz vorgegeben oder kann über die Merkmalsvorauswahl umständlich eingestellt werden. Das bestehende System bietet hier eine höhere Flexibilität und Effizienz, der Ablauf ist nicht derartig stark gebunden.

**Berechtigungssystem für Freigaben und Abweichungen**

Derzeit ist das System der Freigaben und Abweichungen sehr restriktiv und es können z. B. Aufträge nicht ohne die erforderlichen Freigaben abgeschlossen werden. Weiters ist das Rechtesystem so

ausgereift, dass es an jede(e) BenutzerIn angepasst werden kann. Abweichungen bei den Produkten und damit verbundene Entscheidungen können bei B&W über das Modul Ruf abgewickelt werden. Hier funktionieren wieder die Berechtigungsstufen über Name und Passwort. Der SPC Auftrag kann aber auch abgeschlossen werden, wenn im Ruf noch keine Entscheidung getroffen wurde.

**Auswirkungen:**

- Höhere Eigenverantwortung der MitarbeiterInnen an den SPC Stationen, und damit höheres Risiko, dass fehlerhafte Produkte zu den KundInnen gelangen.
- Neue interne Abläufe müssen definiert werden, wie und wann MitarbeiterInnen interne Reklamationen auslösen, welche Daten bei der Eingabe erforderlich sind und welche Berechtigungen gelten sollen.

### Systemverfügbarkeit

Beim bestehenden System können die einzelnen Stationen weiterarbeiten, wenn der Server nicht verfügbar ist. Das hat den Vorteil, dass man den Server kurz für Wartungsarbeiten vom Netz nehmen kann, bzw. Ausfälle kompensiert werden können.

**Status B&W:**

Die B&W Lösung besteht aus einem Application Server und einem Datenbank Server. Fällt eines dieser Systeme aus oder muss wegen Wartungsarbeiten vom Netz genommen werden, so kann nicht an den Clients weitergearbeitet werden.

### 6.1.4 Zusätzlich erforderliche Systeme

**MDE System Client**

Derzeit sind beim CAQ System Bruneck die CAQ und MDE in einem Programm integriert, d. h. alle Daten werden in eine Oberfläche eingegeben, Stammdaten müssen so nur einmal eingegeben werden. Beim Umstieg auf CASQ-IT 9000 ist ein separates MDE Modul erforderlich, das zusätzlich entwickelt werden muss.

**Report aller MDE und SPC Meldungen**

Derzeit hat das System CAQ Bruneck den Vorteil, dass alle Informationen zu einem Produktionsauftrag (SPC und MDE) in einem Bericht angezeigt werden. Mit diesem Instrument ist eine schnelle und lückenlose Rückverfolgung möglich. Es werden nicht nur die Messdaten (SPC), sondern auch alle Prozessinformationen angezeigt. Vor allem bei Reklamationen ist eine schnelle Eingrenzung des Problems möglich.

Dieses Reportsystem, das einen Überblick über die Daten der MDE und der SPC gibt, muss noch geschaffen werden.

**Automatische Gewichtsregelung bei T4**

Im derzeitigen System ist ein Verfahren zur automatischen Nachregelung von T4-Pressen bei Gewichtsabweichungen integriert. Diese Art der Regelung erfordert Echtzeiteigenschaften eines Betriebssystems. Unter MSDOS ist dies möglich, da immer nur ein Prozess (Programm) den Prozessor beansprucht. Moderne Multitasking Betriebssysteme wie Windows oder Unix/Linux sind in der Regel nicht ohne weiteres echtzeitfähig. Hier muss ein neuer Ansatz gefunden werden, um diesen Teil abzudecken. Es wird erforderlich sein, ein neues Konzept für diese Maschine zu entwickeln, da B&W nur eine Schnittstelle bereitstellt und derartige Steuerungen nicht Teil der von ihnen gelieferten Software sind.

**Parameter Kontrollsystem (Öl, Mix)**

Im CAQ System Bruneck werden bei einem Produktionsauftrag verschiedene Prozessinformationen eingegebenen. Einige dieser Parameter werden überprüft, beispielsweise erfolgt eine Kontrolle, ob der richtige Werkstoff oder das richtige Imprägnieröl verwendet wird. Diese Absicherung wurde aufgrund einer Kundenforderung implementiert. Ein System, das diese Parameter überprüft, muss noch geschaffen werden.

### 6.1.5 Vor- und Nachteile

**Anpassung an neue Normen**

Neue Normforderungen werden automatisch in neue Versionen des Systems aufgenommen. Diese sind durch einen Supportvertrag gedeckt.

**Externer Support und Wartung**

B&W verfügt über ein 7/24 Callcenter, mit dem auftretende Probleme besprochen werden können. Da es über einen Zugang zu den Servern verfügt, kann es sich in das System einwählen und so das Problem analysieren.

**Integration der Module**

Die einzelnen Module des CASQ-IT 9000 greifen ineinander, Daten, die in einem Modul eingegeben werden, sind auch in den anderen verfügbar.

**Kontinuierliche Weiterentwicklung**

B&W ist bestrebt, seine Produkte kontinuierlich zu erweitern und an neue Anforderungen anzupassen, die dann allen KundInnen zur Verfügung gestellt werden können.

## 6.1.6 Kosten

| Szenario B&W | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Direkte Kosten** | | | | | | **1. Jahr** | | **2. Jahr** | | **3. Jahr** | | **1-3. Jahr** |
| Client | | | | | Stk. | €/Stk. | € tot | €/Stk. | € tot | €/Stk. | € tot | € tot |
| | **Hardware** | | | | | | | | | | | |
| | | | IBRit-mc4, -di8 | | 70 | 1.300 | 91.000 | 0 | 0 | 0 | 0 | 91.000 |
| | | | PC | | 70 | 1.000 | 70.000 | 0 | 0 | 0 | 0 | 70.000 |
| | | | Microsoft W2K | | 70 | 150 | 10.500 | 0 | 0 | 0 | 0 | 10.500 |
| | | | **Summe:** | | | 2.450 | 171.500 | | 0 | | 0 | **171.500** |
| Server | | | | | | | | | | | | |
| | **Hardware** | | | | | | | | | | | |
| | | | App. Server Dell 2850 | | 1 | 3.700 | 3.700 | 0 | 0 | 0 | 0 | 3.700 |
| | | | DB Server Dell 2850 | | 1 | 3.700 | 3.700 | 0 | 0 | 0 | 0 | 3.700 |
| | **Software Kauf** | | | | | | | | | | | |
| | | | MS Server OS | | 1 | 700 | 700 | 0 | 0 | 0 | 0 | 700 |
| | | | MS Client Lizenz | | 70 | 20 | 1.400 | 0 | 0 | 0 | 0 | 1.400 |
| | | | MKS Toolkit | | 1 | 1.100 | 1.100 | 0 | 0 | 0 | 0 | 1.100 |
| | | | Oracle 9i | | 1 | 6.500 | 6.500 | 0 | 0 | 0 | 0 | 6.500 |
| | | | **Summe:** | | | 15.720 | 17.100 | | 0 | | 0 | **17.100** |
| B&W | | | | | | | | | | | | |
| | **Rahmenvertrag** | | | | | | | | | | | |
| | | | CPL, EMPB, ZP, SPC/QDE, RUF | | 1 | 10.000 | 10.000 | 0 | 0 | 0 | 0 | 10.000 |
| | | | Werkumlage | | 1 | 4.000 | 4.000 | 0 | 0 | 0 | 0 | 4.000 |
| | | | Named User Lizenz | | 150 | 200 | 30.000 | 0 | 0 | 0 | 0 | 30.000 |
| | | | Standard-Servicevertrag Server | | 1 | 6.000 | 6.000 | 6.000 | 6.000 | 6.000 | 6.000 | 18.000 |
| | | | Standard-Servicevertrag Client | | 40.000 | 1,5% | 7.200 | 1,5% | 7.200 | 1,2% | 5.760 | 20.160 |
| | **Erforderliche Dienstleistungen** | | | | | | | | | | | |
| | | | Modultraining | | 6 | 1.200 | 7.200 | 0 | 0 | 0 | 0 | 7.200 |
| | | | Einrichtung auf Server | | 3 | 1.150 | 3.450 | 0 | 0 | 0 | 0 | 3.450 |
| | | | Installation, Test und Inbetriebn. | | 3 | 1.150 | 3.450 | 0 | 0 | 0 | 0 | 3.450 |
| | **Zusatzanforderungen** | | | **Prio.** | | | | | | | | |
| | | 1.11 | Berechtigung für Freigaben | H | 1 | 6.500 | 6.500 | 0 | 0 | 0 | 0 | 6.500 |
| | | 1.12 | Automatische Berichterstellung | H | 1 | 11.500 | 11.500 | 0 | 0 | 0 | 0 | 11.500 |
| | | 1.13 | Bericht Vorlagen | H | 1 | 5.800 | 5.800 | 0 | 0 | 0 | 0 | 5.800 |
| | | 1.15 | Cpk Wert Abweichung | H | 1 | 3.450 | 3.450 | 0 | 0 | 0 | 0 | 3.450 |
| | | 1.16 | Hintergrundmessung (für T4) | H | 1 | 4.000 | 4.000 | 0 | 0 | 0 | 0 | 4.000 |
| | | 1.16 | Projektarbeit | H | 2 | 1.150 | 2.300 | 0 | 0 | 0 | 0 | 2.300 |
| | | 1.16 | Hintergrundmessung (für ZF) | H | 1 | 4.000 | 4.000 | 0 | 0 | 0 | 0 | 4.000 |
| | | 1.16 | Projektarbeit | H | 2 | 1.150 | 2.300 | 0 | 0 | 0 | 0 | 2.300 |
| | | 1.16 | Remote Prüfung | H | 1 | 9.000 | 9.000 | 0 | 0 | 0 | 0 | 9.000 |
| | | 1.16 | Projektarbeit | H | 2 | 1.150 | 2.300 | 0 | 0 | 0 | 0 | 2.300 |
| | | 1.2 | Prüfintervalle pro Stück | M | 1 | 9.000 | 9.000 | 0 | 0 | 0 | 0 | 9.000 |
| | | 1.2 | Projektarbeit | H | 2 | 1.150 | 2.300 | 0 | 0 | 0 | 0 | 2.300 |
| | | 1.7 | Versionsmanagement | M | 1 | 9.000 | 9.000 | 0 | 0 | 0 | 0 | 9.000 |
| | | 1.3 | Bei Stillstand Zeitbalk. Stop | L | 1 | 5.900 | 5.900 | 0 | 0 | 0 | 0 | 5.900 |
| | | 1.3 | Projektarbeit | H | 2 | 1.150 | 2.300 | 0 | 0 | 0 | 0 | 2.300 |
| | **Spracherweiterungen** | | | | | | | | | | | |
| | | | Sprache pro Server | | 2 | 10.000 | 20.000 | 0 | 0 | 0 | 0 | 20.000 |
| | | | Sprache pro Modul | | 10 | 2.500 | 25.000 | 0 | 0 | 0 | 0 | 25.000 |
| | | | **Summe:** | | | | 195.950 | | 13.200 | | 11.760 | **220.910** |
| Zusatzsysteme | | | | | Mo | €/m | | | | | | |
| | | | MDE Client Software | | 1 | 16.000 | 16.000 | 0 | 0 | 0 | 0 | 16.000 |
| | | | Parameter Kontrollsystem | | 1 | 16.000 | 16.000 | 0 | 0 | 0 | 0 | 16.000 |
| | | | Reporting aller SPC/MDE Meldungen | | 1 | 16.000 | 16.000 | 0 | 0 | 0 | 0 | 16.000 |
| | | | T4 Gewichtsmessung + Hardware | | 2 | 16.000 | 32.000 | 0 | 0 | 0 | 0 | 32.000 |
| | | | **Summe:** | | | | 80.000 | | 0 | | 0 | **80.000** |
| **Indirekte Kosten** | | | | | | **1. Jahr** | | **2. Jahr** | | **3. Jahr** | | **1-3. Jahr** |
| Allgemeine Kosten | | | | | | | | | | | | |
| | **Adaptierung** | | | | | | | | | | | |
| | | | Arbeitsplatz | | 70 | 250 | 17.500 | 0 | 0 | 0 | 0 | 17.500 |
| | | | **Summe:** | | | 250 | 17.500 | 0 | 0 | 0 | 0 | **17.500** |
| | | | **Summe:** | | | | **482.050** | | **13.200** | | **11.760** | **507.010** |
| Zusätzliche Projektarbeit Schätzung (1150€/Tag) | | | | Basisbetrag, % Angaben pro Monat, siehe Angebot B&W | | | | | | | | |

**Tabelle 6: Kostenaufstellung Szenario B&W**

### 6.1.7 Anmerkung zu Böhme und Weihs

Grundsätzlich muss man zu Böhme und Weihs anmerken, dass Termine kaum eingehalten, Fristen ohne Reaktion verabsäumt oder Termine verschoben wurden. Das lässt auf mangelndes Interesse am Auftrag oder permanente Überlastung schließen.

## 6.2 Szenario Reengineering

Hier wird jenes Szenario skizziert, wenn auf Basis der bestehenden Software eine Eigenentwicklung durchgeführt wird, die einen Großteil der Abläufe und Anforderungen bereits definiert.

### 6.2.1 Allgemeine Szenariobeschreibung

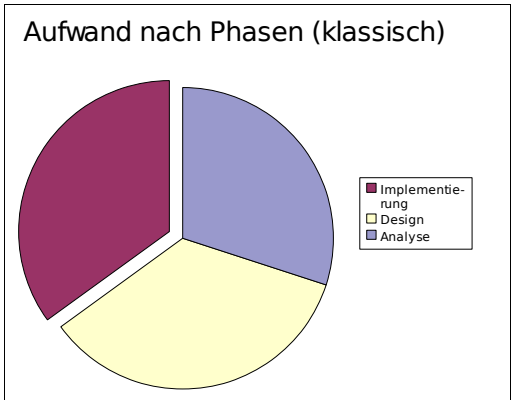


Abbildung 29: Aufwand in der Softwareentwicklung

Neue Softwareprojekte werden in der Regel durch ein klassisches Forward Engineering abgewickelt. Ausgehend von der Analyse der Anforderungen wird in der Designphase der logische Aufbau entwickelt und dann in den Programmcode implementiert. Die Implementierungsphase (Programmierung) hat dabei einen Anteil von unter 40 % bei gängigen Software Projekten. Viel mehr Aufwand steckt in der Spezifikation und im Design der Software.

Sind nun bestehende Softwareprojekte vorhanden, können Teile des Designs und der Abläufe übernommen werden und müssen so nicht vollständig neu entwickelt werden. D. h., es sind viele Informationen in Form von Daten, Algorithmen und Abläufen bereits vorhanden und können wiederverwendet werden. Somit ist eine breite Basis vorhanden, die weiterentwickelt und mit neuen, zusätzlichen Anforderungen bereichert wird. Da im Zuge des Projekts bereits ein Reverse Engineering durchgeführt wurde, kann dies als Basis für ein Forward Engineering verwendet werden.

Gegen eine proprietäre Lösung wird häufig mit dem Argument der Inkompatibilität und den vermeintlich höheren Kosten argumentiert. Da der Markt für CAQ Systeme breit gefächert ist, lässt sich das Argument der Inkompatibilität leicht entkräften, da alle Systeme Schnittstellen bereitstellen müssen, um am Markt zu überleben. D. h. diese Schnittstellen sind vorhanden und können bereits verwendet werden. Somit ist es auch möglich, eine gesunde Mischung zu finden, zwischen Standardsoftwaremodulen, die für viele Aufgaben reichen, und proprietären Modulen, die die unternehmenspezifischen, komplexen Bereiche abdecken. Bei diesem Projekt ist bereits eine solche Mischung aus verschiedenen Modulen vorhanden.

### 6.2.2 Vorgehensweise und Aufbau

Bei einer Migration kann man viele Wege gehen und jeder einzelne wird auch zum Ziel führen.

Eine hohe Flexibilität zeichnet wohl jeden dieser Wege aus. Eine mögliche eigene Lösung kann in folgenden Schritten vollzogen werden bzw. aus folgenden Bereichen bestehen: (Im frühen Stadium der Planung kann keine exakte Kostenschätzung erwartet werden.)

1. Restructuring der Datenbank auf ein RDBMS (Oracle 10i)
2. Migration der Daten
3. Reengineering der Komponenten (SPC, Prüfplan, Abweichungskarten)
4. Hinzufügen neuer Anforderungen
5. Schnittstellen bereitstellen (Messgeräte, BaaN, T4 etc.)
6. Integration und Test

Der zeitliche Aufwand kann nicht ohne genauere Spezifikation der neuen Zusatzanforderungen angegeben werden.

Leitlinie einer eigenen Lösung muss die Verwendung des bestehenden Wissens und die Realisierung mittels Standardkomponenten sein. Zu dem Zeitpunkt, als das derzeitige System entwickelt wurde, musste die gesamte Benutzeroberfläche und die grafischen Ansichten erstellt werden. Mittlerweile gibt es eine Menge an Standardkomponenten, die diese Arbeit zum Teil überflüssig machen. Diese können entsprechend genutzt werden.

### 6.2.3 Vor- und Nachteile

**Übernahme der bestehenden Anforderungen**
Alle Anforderungen, die das System derzeit abbildet, können, wenn eine Sinnhaftigkeit besteht, in ein neues System übernommen werden.

**Wissenserhalt**
Im Laufe der Jahre wurde viel Wissen und Erfahrung in dieses System integriert. Eine eigene Lösung hat einen entscheidenden Vorteil gegenüber einer Standard Software. Sie kann dieses Wissen

mitnehmen und durch neue, unternehmensspezifische Anforderungen bereichert werden.

**Integration in bestehende Systeme**

Jede IT-Infrastruktur eines Unternehmens ist unterschiedlich. Viele Standardsysteme sind Insellösungen: Sie decken ihren Bereich ab, aber es ist schwierig, sie in bestehende Strukturen zu integrieren. Fragen zu Schnittstellen und Integration in andere Systeme können leichter gelöst werden, da nicht zwei Systeme von verschiedenen Fremdherstellern abgestimmt werden müssen.

**Geringer Schulungsaufwand**

Da sich das neue System am Bestehenden orientiert, findet sich der/die BenutzerIn leichter zurecht. Z.B. können auch bestehende Tastenkombinationen weitgehend übernommen werden.

**Kooperation interner und externer Ressourcen**

Auch wenn die Entwicklung nach außen vergeben wird, können für Anpassungen bzw. Erstellung der benötigten Schnittstellen interne Ressourcen verwendet werden. Das Wissen über die bestehende IT Infrastruktur kann schnell und effizient in diese Schnittstellen einfließen.

Werden interne MitarbeiterInnen an der Entwicklung beteiligt, bleibt das Wissen über das System im Unternehmen und allfällige Probleme können intern bewältigt werden.

**Verwendung der bestehenden Hardware**

Die derzeit verwendete Hardware ist solide gebaut. In 15 Jahren wurde noch keine der Interface Boxen ausgetauscht, insgesamt wurden 4 bis 5 Netzteile getauscht und einige Relais. Die Reparatur beschränkte sich in 80 % der Fälle auf durchschnittlich 50 Euro und 1 Stunde Arbeit in einem Zeitraum von 5 Jahren. Diese Hardware kann weiterverwendet werden.

**Datenübernahme**

Da sich das neue System am bestehenden orientiert, ist es möglich, die vorhandenen Daten zu übernehmen. So sind alle Daten des alten Systems im neuen enthalten. Das hat den Vorteil, dass alle Daten in einem System gefunden werden und dass man sichergehen kann, dass das neue System das bestehende vollständig abdeckt.

**Neue Möglichkeiten**

In einer eigenen Lösung können viel mehr neue Technologien eingesetzt werden, die neue Möglichkeiten bieten:

- Nachrichtensystem zur Kommunikation zwischen den Stationen
- Freigaben unabhängig vom Standort
- etc.

**Veränderung der Anforderungen**

Neue Anforderungen, die sich aus Normen ergeben, müssen als Change Request in das System aufgenommen und implementiert werden.

**Lokale Lösung**

Kein werkübergreifendes System, Prozesse und Abläufe können schwer über das Werk hinaus verglichen, Vorteile des Systems können in den anderen Werken nur zum Teil umgesetzt werden. Diese Software ist sofort für die Werke Bruneck, Mailand und Kolsvar einsetzbar. Ein Einsatz in anderen Werken ist nur auf Basis von standardisierten Abläufen sinnvoll.

**Keine kontinuierliche Weiterentwicklung**

Alle zukünftigen Änderungen, die an das System gestellt werden, müssen als Change Request aufgenommen werden und kostenpflichtig implementiert werden.

### 6.2.4 Kosten

| Szenario Reengineering | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Direkte Kosten** | | | | | **1. Jahr** | | **2. Jahr** | | **3. Jahr** | **1-3. Jahr** |
| Client | | | | Stk. | €/Stk. | € tot | €/Stk. | € tot | €/Stk. | € tot | € tot |
| | **Hardware** | | | | | | | | | | |
| | | Client PC | | 70 | 1.000 | 70.000 | 0 | 0 | 0 | 0 | 70.000 |
| | | | **Summe:** | | 1.000 | 70.000 | | 0 | | 0 | **70.000** |
| Server | | | | | | | | | | | |
| | **Hardware** | | | | | | | | | | |
| | | App. Server Dell 2850 | | 1 | 3.700 | 3.700 | 0 | 0 | 0 | 0 | 3.700 |
| | | DB Server Dell 2850 | | 1 | 3.700 | 3.700 | 0 | 0 | 0 | 0 | 3.700 |
| | **Software Kauf** | | | | | | | | | | |
| | | Oracle 9i | | 1 | 6.500 | 6.500 | 0 | 0 | 0 | 0 | 6.500 |
| | | | **Summe:** | | 13.900 | 13.900 | | 0 | | 0 | **13.900** |
| Entwicklungskosten Reengineering | | | | | | | | | | | |
| | **Datenbank und Applikationsserver** | | | Mon. | €/m | | | | | | |
| | | Datenmigration | | 1 | 10.000 | 10.000 | 0 | 0 | 0 | 0 | 10.000 |
| | | Datenbank | | 3 | 10.000 | 30.000 | 0 | 0 | 0 | 0 | 30.000 |
| | | Applikationsserver Software | | 3 | 10.000 | 30.000 | 0 | 0 | 0 | 0 | 30.000 |
| | **Client Software** | | | | | | | | | | |
| | | SPC | | 4 | 10.000 | 40.000 | 0 | 0 | 0 | 0 | 40.000 |
| | | Prüfplanung | | 2 | 10.000 | 20.000 | 0 | 0 | 0 | 0 | 20.000 |
| | | Fehlermanagement | | 2 | 10.000 | 20.000 | 0 | 0 | 0 | 0 | 20.000 |
| | **Installation und Test** | | | | | | | | | | |
| | | System und Integrationstest | | 1 | 10.000 | 10.000 | 0 | 0 | 0 | 0 | 10.000 |
| | **Prüfmittelmanagement CASQ-IT 9000** | | | | | | | | | | |
| | | Client Lizenzen | | 4 | 203 | 812 | 0 | 12 | 0 | 10 | 834 |
| | | Server Beteiligung | | 1 | 609 | 609 | 0 | 9 | 0 | 7 | 625 |
| | | | **Summe:** | | | 161.421 | | 21 | | 17 | **161.459** |
| | | | **Summe:** | | | **245.321** | | **21** | | **17** | **245.359** |

**Tabelle 7: Kostenabschätzung Szenario 2 Reengineering**

## 6.3 Vergleich beider Szenarien

Hier werden noch einmal die markanten Unterschiede beider Szenarien herausgearbeitet und tabellarisch dargestellt:

### 6.3.1 Hardware

| | Szenario Reengi-neering | Szenario B&W |
|---|---|---|
| Messbo- | Wiederverwendung | Neue erforderlich |

| | Szenario Reengineering | Szenario B&W |
|---|---|---|
| xen | | |
| Ausführung | Robust ausgereift | Flexibles System aus mehreren Boxen |
| Reparatur | Intern ca. 50€ / 5 Jahre | Extern |
| Kosten für Bruneck | ca. 85.000€ | ca. 200.000€ |

**Tabelle 8: Szenarienvergleich Hardware**

### 6.3.2 Software

| | Szenario Reengineering | Szenario B&W |
|---|---|---|
| | Flexibilität | Standard |
| Funktionsumfang | Nur benötigte Funktionen<br>Alle vom bestehenden System werden übernommen | CASQ-IT 9000 großer Funktionsumfang, da Standardsoftware<br>Zusatzsysteme erforderlich für nicht gedeckte Anforderungen |
| Benutzeroberfläche Messstation | All in one Oberfläche<br>Orientierung am CAQ-Bruneck | Mindestens 2 Applikationen<br>CASQ-IT 9000 mit Modulen + Zusatzsystemen |
| Wartung/Support | Intern | 24/7 Callcenter über Wartungsvertrag |
| Ausfallsicherheit | Wenn der Fileserver ausfällt, können die Clients weiterarbeiten, Zwischenspeicherung der Daten | Server bilden „single point of failure"<br>Fällt ein Server aus oder muss wegen Wartungsarbeiten vom Netz genommen werden, arbeiten die Clients NICHT weiter |
| Kosten | ca. 200.000€ | ca. 300.000€ |

**Tabelle 9: Szenarienvergleich Software**

### 6.3.3 Rollout

Betrachtet man die direkten Projektkosten und den Aufwand für die Schulung, so sind die Vorteile des einen oder anderen Systems auf den ersten Blick nicht wirklich ersichtlich. Man muss aber dazusagen dass durch ein Reengineering bestehende Abläufe sofern sinnvoll beibehalten werden und eine Umschulung leichter fällt.

| | **Szenario Reengi-neering** | **Szenario B&W** |
|---|---|---|
| Schulungs-aufwand | Gering, da Abläufe und Strukturen er-halten bleiben | 300 Mitarbeiter müssen auf neue Abläufe und Benutzeroberflächen geschult werden |
| Change Ef-fekte | niedrig | hoch |

**Tabelle 10: Szenarienvergleich Rollout**

# 7 Zusammenfassung und Ausblick

Aufgrund der prinzipiellen Heterogenität der beiden diskutierten Lösungen ist ein direkter Vergleich basierend auf Tabellen und Zahlen nicht aussagekräftig. Das Szenario eines Reengineering des bestehenden Systems steht dem Szenario eines Ankaufs einer Standardsoftware mit entsprechendem Customizing gegenüber.

Es wäre zu ungenau, einfach nur die Kosten und die Anforderungen an die Programme an sich heranzuziehen und für eine Entscheidungsfindung zu verwenden.

Viel wichtiger ist es, die beiden Systeme, die durch die Szenarien ermittelt wurden, in ihrem Umfeld zu betrachten und die Möglichkeiten für die weitere Entwicklung in der Zukunft miteinzubeziehen. Es hat sich gezeigt, dass die Möglichkeiten für das weitere Vorgehen durch das Entwerfen und Verfeinern von Szenarien klarer werden und so die Vor- und Nachteile herausgearbeitet werden können. Die beiden finalen Szenarien, die hier erarbeitet wurden, sind aus einer Vielzahl von Lösungswegen entstanden.

Es war essenziell die gesamte Menge an Möglichkeiten zu betrachten, die es zur Lösung der Grundanforderung gibt, nämlich die Qualität im Unternehmen zu garantieren und zu sichern. Von der Vielzahl der Wege, die in Betracht gezogen wurden, wurde jeder auf seine Machbarkeit hinsichtlich Zeit und Kosten bewertet und darauf geachtet, dass dieser Lösungsvorschlag das Projekt insgesamt auch schneller und effizienter ans Ziel führt. Viele erstklassige Teillösungen wurden aus unterschiedlichsten Gründen wieder verworfen, weil sie insgesamt keinen Vorteil für das Projekt als Ganzes darstellten oder aus der Erfahrung des Unternehmens bekannt war, dass ein Weg sich zwar im ersten Moment hervorragend darstellt, sich aber ähnliche

Wege sich in der Vergangenheit zu einem Boomerang entwickelt haben und schlussendlich kontraproduktiv waren.

Die Arbeit baut im Wesentlichen auf zwei theoretischen Säulen auf, die beide in den Eingangskapiteln behandelt werden: einerseits auf dem Qualitätswesen in einem Unternehmen, hier vor allem auf dem operativen Teil, und zum anderen auf der Legacy Systems Problematik.

Es wäre fatal gewesen, das Problem nur von der einen oder anderen Seite zu betrachten. So hätte man beide Male das Ziel aus den Augen verloren und entweder eine rein technische Lösung durch ein Reengineering des Systems herbeigeführt oder eine Lösung mit dem bereits vorhandenen IT Potential, das noch im alten System steckt, nicht genutzt.

Die Grundlage für das Erstellen der Szenarien bot die technische Analyse des bestehenden Systems. Bei diesem Reverseegineering wurde versucht, das bestehende System in seinen Datenstrukturen und Abläufen zu erfassen und zu dokumentieren. Da der bestehende Code nur noch schwer verständlich und schlecht dokumentiert war, war es sinnvoll, das System als Black Box zu betrachten. So wurden durch das Apprenticing, das „in die Lehre gehen", einige Details ans Licht gebracht, die man aus dem Programmcode kaum hätte herauslesen können.

Außerdem haben sich auch die Schwächen des bestehenden Systems gezeigt und daraus konnten bereits neue Anforderungen abgeleitet werden, was eigentlich erst in der nächsten Projektphase erfolgt wäre.

Am Ende der Analyse ist ein Katalog von bestehenden Anforderungen entstanden, bereichert mit neuen Punkten. Auf Basis dieses Dokuments wurden dann die Szenarien ausgearbeitet.

Anfangs wurden drei Szenarien verfolgt, der Einsatz von reiner Standardsoftware, ein Reenineering oder eine Mischung aus beidem. Das Szenario, in dem Standardsoftware, bereichert mit eigenen Komponenten eingesetzt wird, wurde schnell verworfen, da es von der Herstellerseite nicht ausreichend unterstützt wurde.

Die beiden anderen, das Reengineering des alten Systems unter Berücksichtigung der neuen Anforderungen, sowie der Ankauf einer reinen Standardsoftware, wurden dann weiter ausgearbeitet. Dabei wurden vermeintlich unwichtige Details entdeckt, die sich genau als jene Punkte entpuppten, die ein Projekt zu Fall bringen können. Schlussendlich kristallisierten sich dann zwei Szenarien heraus, die es beide ermöglichen würden, das alte System durch ein neues zu ersetzen. Und trotzdem gab es am Ende nicht zwei gleichwertige Gewinner.

Mit der Lösung von B&W hätte man die weitere Entwicklung des Systems aus der Hand gegeben. Man kann davon ausgehen, dass es den Anforderungen für die Qualitätssicherung nach ISO 9000 und ISO -TS16949 auch in Zukunft gerecht geworden wäre und dass es Änderungen in der Norm schnell möglich gemacht hätte.

Dieses System hätte aber nicht ausgereicht, um das derzeitige zu ersetzen.

Das System von B&W muss also erweitert und neue Systeme und Schnittstellen geschaffen werden. Der Vorteil dabei ist, dass man intern kein Wissen über das System benötigt, sondern Support und Weiterentwicklung einfach auslagern kann. Neue, zukünftige Anforderungen müssen, was Preis und Machbarkeit betrifft, dann extern mit B&W verhandelt werden.

Der Hersteller hatte aber vor allem ein Ziel: Sein System ohne große Anpassungen zu verkaufen, komplexe Anforderungen wurden gar nicht unterstützt oder es wurde versucht, sie durch komplexes

Customizing über Umwege im System abzubilden. Das Resultat wäre also ein effektives, aber kein effizientes System gewesen. Macht man sich die Folgen dieser Tatsache bewusst, wird klar, dass eine reine Betrachtung der direkten Kosten vollkommen unzureichend ist, denn ein effektives aber ineffizientes System für 400 MitarbeiterInnen wird zwar die Grundanforderung „Sichern der Qualität" erfüllen, aber schnell einen höheren Personalaufwand nach sich ziehen. Und sei es „nur" zwei MitarbeiterInnen mehr in der Produktion. Das mag im Verhältnis zu den 400 als gering angesehen werden, aber vergleicht man die jährlichen Kosten, die für zwei Facharbeiter anfallen, dann amortisiert sich eine effiziente Lösung – auch wenn sie teuer erscheint – innerhalb kurzer Zeit.

Zudem hat die eigene Lösung den Vorteil, dass sie alle bestehenden Anforderungen übernimmt und besser in die vorhandene IT-Infrastruktur integriert werden kann. Das Wissen über das System bleibt im Unternehmen und kann so auch weiterentwickelt werden. Der Nachteil ist, dass Änderungen der Qualitätsnormen im System nachgezogen werden müssen. Bei B&W wäre dies durch den Supportvertrag abgedeckt. Dafür bietet diese Lösung eine höhere Flexibilität und Integration in die bestehende IT Landschaft. Dies ist ein entscheidender Faktor, der kaum mit Zahlen beziffert werden kann.

Alles in allem wurde die Arbeit mit dem Ziel begonnen, einen Weg zu finden, wie man ein bestehendes System, das in einem Unternehmen integriert ist, durch ein neues ersetzt. Dabei war es nötig, nicht nur das System an sich zu betrachten, sondern auch die Grenzen und die Integration in andere Systeme, sowie die Akzeptanz in der Belegschaft.

Das Ergebnis waren zwei Szenarien, die den ursprünglichen Forderungen genügen. Aber wirft man einen Blick über den

Tellerrand, stellt man fest, dass ein effektives Erfüllen der Anforderungen allein nicht zielführend ist. Effizienz bleibt allzu leicht auf der Strecke – und genau darin liegt das Verbesserungspotential.

Dieses System ist für die Abwicklung von Kernprozessen des Unternehmens unerlässlich, eine kostengünstige Sicherung der Qualität kann für die Position am Markt von entscheidender Bedeutung sein. Hier Standardsoftware zu verwenden, bedeutet, sich so zu verhalten wie andere Marktteilnehmer. Ein bestehendes System weiterzuentwickeln, birgt das Risiko, dass das System schlechter wird als die Standardsoftware; doch es kann verbessert werden und einen entscheidenen Wettbewerbsvorteil bringen. Die Anforderungen, die an das über 15 Jahre alte System gestellt wurden, konnten von der aktuellen Standardsoftware noch immer nicht erfüllt werden und alle ProjektteilnehmerInnen sprachen von einem Rückschritt, sollte dieses System Verwendung finden. Über die Jahre wurden viele Verbesserungen eingearbeitet. Diese würden verloren gehen, da sie in die Standardsoftware nicht übernommen werden können, weil sie zu unternehmensspezifisch sind.

Als Resultat der Arbeit ist die Empfehlung den Weg der vergangenen Jahre fortzuführen, und somit ein Reengineering des bestehenden Systems durchzuführen, um einen Wettbewerbsvorteil in der Qualitätssicherung zu erhalten.

# Literaturverzeichnis


[1]ISO 8402:2000 QUALITY MANAGEMENT AND QUALITY ASSURANCE - VOCABULARY

[2]ISO/TS 16949:2002Qualitätsmanagementsysteme, Besondere Anforderungen bei Anwendung von ISO 9001:2000 für die Serien- und Ersatzteilproduktion in der Automobilindustrie

[3]www.qualitainer.de

[4]QS-9000FMEA - Fehler-Möglichkeits- und -Einfluss-Analyse, 3. Aufl. 10.2001, Carwin Ltd.

[5]DIN EN 60812Analysetechniken für die Funktionsfähigkeit von Systemen - Verfahren für die Fehlzustandsart- und -auswirkungsanalyse

[6]Dietrich, E/Schulze, AQualitätsregelkarten und Fähigkeitsindizes bilden eine Einheit. Berechnung der Fähigkeitsindizes im Vergleich1992

[7]Mann, H;Schiffelgen, HEinführung n die Regelungstechnik, 6. Aufl.

[8]VDA-QMCAusgabe 2003 8d-Beschreibung

[9]DGO, Deutsche Gesellschaft für Qualität, Arbeitsgruppe 136: Prüfmittelüberwachung Frankfurt 1994

[10]Dietrich;SchulzeStatistische Verfahren zur Qualifikation von Messmitteln, Maschinen und Prozessen, 3 Auflage1998

[11]Dietrich, E.; Schulze, A.Geeignete Verteilungsmodelle finden1994

[12]PSA Peugeot, Citroen, RenaultProduktionsmittel, Zulassung der Funktionsfähigkeit von Messmitteln, Spezifische Prüfmittel.1991

[13]DGO Schrift Nr. 18-18Anleitung zur Qualitätsregelkarte und zur Fehlersammelkarte1990

[14]IEEE Standards of software maintainance1998

[15]Chikofsky, E.J. , Cross, J.H.reverse enineering und desing recovery: a taxonomy1990

[16] M. Fowler et al., Refactoring: Improving the Design of Existing Code, Addison Wesley Longman, Reading, Mass., 1999.

[17]Ramage, Magnus, Carole Brooke, Keith Bennett and Malcolm Munro.Evaluating solutions to legacy systems. Proceedings of the 6th European Conference on the Evaluation of IT, 4-5 November 1999, Uxbridge, UK.

[18]M.M. LehmanPrograms, Life-Cycles, and Laws of Program Evolution1980

[19]A. De Lucia, A. R. Fasolino, E. PompellaA Decisional Framework for Legacy System Management

[20]K.H. Bennett, M. Ramage and M. MunroDecision model for legacy systems1999

[21]Brooke, Carole, Magnus Ramage, Keith Bennett and Nicolas GoldChange effects and legacy information systems: a framework to aid our understanding. Pro-

ceedings of the IFIP working groups 8.2 and 8.6 joint working conference on Information Systems: Current Issues and Future Changes, Helsinki, 10-13 December 1998.

[22]A.Bianchi, D. Caivano, G. VisaggioIterative Reenigneering of Legacy Systems,IEEE Transactions on Software Engineering Vol. 29 No3. March 2003

[23]A. Bianchi, D. Caivano, G. Visaggio"Method and Process for Iterative REengineering Data in Legacy System" Proc. IEE Working Conf. REv. Eng. pp. 86-96 2000

[24]A. Bianchi, D. Caivano, V. Marengo, G. Visaggio"Iterative REengineering of Legacy Functions" Proc. IEEE Int Conf. Software Maintenence pp 632-641, 2001

[25] G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide. Addison Wesley, 1999.

# CAQ System Anbieter

| Quelle<br><br>www.qualitainer.de | Prüfplanung | Wareneingangsprüfung | Warenausgangsprüfung | Fertigungsprüfung/ SPC | Prüfmittelmanagement | Prüfmittelfähigkeit/MSA | Reklamationsmanagement | FMEA | APQP | Controlplan | Erstmusterprüfberichte | Dokumentenmanagement | Auditmanagement |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CAQ Bruneck** | **x** | | | **x** | | | **x** | | | **x** | | **x** | |
| ACE Consulting & Engineering GmbH | | | | | | | | | | | x | | |
| Aguntum Gesellschaft für Softwareentwicklung | x | x | x | x | x | x | x | | | | x | | |
| AHP Gesellschaft für Informationsverarbeitung... | x | x | x | x | x | x | x | x | x | x | x | | x |
| aim-systems GmbH | | | | | | | | | | | | x | |
| APIS Informationstechnologien GmbH | | | | | | | | x | x | x | | | x |
| Babtec GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| Böhme & Weihs | x | x | x | x | x | x | x | x | x | x | x | x | x |
| CAQ AG Factory Systems | x | x | x | x | x | x | x | x | x | x | x | x | x |
| CAT GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| COMPASS | | | | | | | | | | | | x | |
| ConSense GmbH | | | | | | | | | | | | x | |
| Consic Software Engineering | x | | | x | x | | | | | | | | |
| DATA INPUT AG | x | x | x | x | | | x | | x | x | x | | x |
| Dato(HM) - CAQ-TOOLS | x | | | x | x | | x | x | | | x | x | x |
| ddw Computersysteme GmbH | x | x | x | x | x | x | x | x | x | x | x | x | |
| DOCOSYS GmbH | | | | | | | | | | | | x | |
| DORER MANAGEMENTSYSTEME | | | | | | | | | | | | x | |
| Dr. Binner CIM-house | | | | | | | | x | | | | x | x |

| Quelle<br><br>www.qualitainer.de | Prüfplanung | Wareneingangsprüfung | Warenausgangsprüfung | Fertigungsprüfung/ SPC | Prüfmittelmanagement | Prüfmittelfähigkeit/MSA | Reklamationsmanagement | FMEA | APQP | Controlplan | Erstmusterprüfberichte | Dokumentenmanagement | Auditmanagement |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| GmbH | | | | | | | | | | | | | |
| Elias GmbH | x | | | x | | | | | | | | | |
| EVO Informationssysteme GmbH | x | x | x | x | x | | x | x | | | x | | |
| GEWATEC GmbH & Co. KG | x | x | x | x | x | x | x | x | x | x | x | x | x |
| GUARDUS Solutions AG | x | x | x | x | x | x | x | x | x | x | x | x | x |
| Hubert Dix Team Computing | | | | | | | x | | | | | x | |
| IB SETEQ GmbH | x | x | x | x | x | x | x | | | | x | x | |
| IBS AG | x | x | x | x | x | x | x | x | x | x | x | x | x |
| IBS SINIC GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| IMCOR GmbH | | | | | x | | | x | | | | x | |
| innotec Systemtechnik GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| InQu Informatics GmbH | x | x | x | x | x | x | x | x | x | x | x | | x |
| iqs Software GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| JessenLenz GmbH | x | x | x | x | x | x | x | x | | | x | x | |
| Kistner Messtechnik GmbH | x | x | x | x | x | x | x | x | | | | | |
| LVQ GmbH Chemnitz | x | x | x | x | x | | x | x | | | x | | x |
| MARTIN STUMPE AG | | | | | | | | | | | | x | |
| MESAS Gesellschaft für Mess-, | x | | | x | | | | | | | | | |
| Minks Management Software | | | | | | | | | x | x | | x | x |
| MPDV Mikrolab GmbH | x | x | x | x | x | x | x | | x | x | x | | x |
| ONTIV GbR | | | | | | | | | | | | x | |
| Pickert & Partner GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| PLATO AG | | | | | | | x | x | x | x | | x | x |
| pluscode OHG | | | | | x | | x | | | | | | |
| Q-DAS GmbH & Co. KG | x | | | x | | x | | | | | x | | |

| Quelle<br><br>www.qualitainer.de | Prüfplanung | Wareneingangsprüfung | Warenausgangsprüfung | Fertigungsprüfung/ SPC | Prüfmittelmanagement | Prüfmittelfähigkeit/MSA | Reklamationsmanagement | FMEA | APQP | Controlplan | Erstmusterprüfberichte | Dokumentenmanagement | Auditmanagement |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| QDaTech GmbH | x | x | x | x | x | x | x | | | | x | | |
| Qualiteasy | | | | | | | | | | | | x | x |
| QUIPSY QUALITY GmbH | x | x | x | x | x | x | x | x | x | x | x | x | x |
| QUMsult GbR | | | | | | | | | | | | x | |
| R&K Risk Management, Dipl.-Ing.Peter Richter | | | | | | | | | | | | x | x |
| Rossmanith GmbH | x | x | | x | x | x | x | | | | x | x | x |
| SCIIL | x | x | x | x | | | x | | | | | | x |
| softX eSolutions GmbH | | | | | | | x | | | | | | |
| SYNPROVIS GmbH | | | | | | | x | | | | | | |
| Wille System GmbH | x | x | x | x | x | x | | | | | | | |

**Tabelle 11: Übersicht CAQ Anbieter**