

1 Vorwort

Was war der Auslöser für dieses Buch?

- Aufwandschätzungen liegen regelmäßig zu niedrig. Warum?
- Nicht falsch, sondern das Falsche geschätzt? Warum?
- Anforderungen waren nicht vollständig. Warum?
- Es wird geschätzt und nicht kalkuliert. Warum?

- Anwenderanforderungen sind zu zeitaufwendig in eng gesetzten Terminen.
- Verständigungsschwierigkeiten zwischen Auftraggeber und Auftragnehmer.
- Keine Norm für die Darstellung der Anwenderanforderungen.
- Es fehlt der organisatorische und betriebswirtschaftliche Rahmen für abgesicherte Kalkulationen.

- Wie lösen die Kollegen in anderen Bereichen ähnliche Probleme?
- Mit welchen Methoden und Normen lösen die ihre Probleme?

Für den Leser wird von Interesse sein zu erfahren, warum der Autor sich gerade mit der Thematik des vorliegenden Buches beschäftigt. Was führte in dazu, sich Gedanken zu machen, über eine Methode, die helfen soll, die Softwareentwicklung nachhaltig zu verbessern?

In den vielen Jahren meiner Tätigkeit im Softwarebereich und vor allem im Zusammenhang mit der Software Metrik waren in den unterschiedlichsten Unternehmen immer wieder die gleichen Phänomene zu beobachten. Die wichtigsten waren:

- Die Aufwandschätzungen liegen regelmäßig zu niedrig.
- Es wird nicht falsch, sondern das Falsche geschätzt.
- Die Anforderungen sind nicht vollständig.
- Es wird geschätzt und nicht kalkuliert.

Als guter Organisator nimmt man auffällige Phänomene nicht nur zur Kenntnis. Sondern fragt auch nach deren Ursachen. Auf meine Fragen erhielt ich mit großer Häufigkeit folgende Antworten:

- Die Termine in unseren Projekten sind so eng gesetzt, daß wir für die Erstellung der Anwenderanforderungen nicht genügend Zeit aufwenden konnten.

- Die Verständigung zwischen Anwender und Entwickler ist so schwierig, daß jede Menge Fehlinterpretationen auftraten.
- Fehlende Normen für die Darstellung der Anwenderanforderungen erschweren die Verständigung zwischen allen Beteiligten.
- Einheitliche Kalkulationen werden nicht gefordert, zumal der organisatorische und betriebswirtschaftliche Rahmen für abgesicherte Kalkulationen nicht gegeben ist.

Dem Organisator, der in Unternehmen unterschiedlicher Branchen tätig wurde, fragt man sich dann, ob die Methoden und Normen, mit denen es dort gelingt zuverlässige Kalkulationen zu erstellen, nicht auch auf die Entwicklung von Softwareprodukten übertragen werden kann.

2 Von anderen lernen.

Von Anderen lernen.

- Industrielle Produkte und Softwareprodukte sind sehr unterschiedlich.
 - Materiell
 - Immateriell / abstrakt
- Industrielle Fertigungsplanung beschreibt materielle Produkte *abstrakt*.
- Softwareentwickler beschreiben abstrakte Produkte *abstrakt*.

- Beschreibungsmittel in der Industrie:
 - Stückliste
 - Montageanweisung/Bauanleitung

Glauht man der Mehrzahl der Informatiker, so sind ihre Tätigkeit und deren Gegenstand von ganz besonderer Art. In der Tat sind die Produkte eines Kühlschranksbauers und eines Informatikers nicht vergleichbar. Das eine ist materiell und das andere immateriell.

Vergleicht man dagegen die Prozesse, wie die Produkte und deren Herstellung beschrieben werden, so können/müssen doch wesentliche Parallelen festgestellt werden.

Jede Beschreibung von Produkten und den Produktionsprozesse erfolgt immer auf einer abstrakten, verbalen und oder optischen Ebene. Dabei spielt es keine Rolle, ob es sich dabei um ein Fahrrad oder einen Düsenjet handelt. In der Industrie werden dazu im Wesentlichen zwei verschiedene Beschreibungen genutzt:

Die Stückliste, benennt und zählt die Teile auf, die für ein Produkt benötigt werden.

Die Bauanweisung bzw. die Montageanleitung beschreiben wie und in welcher Reihenfolge die Teile zusammenzubauen sind.

In der industriellen Fertigung wurde die zentrale Bedeutung von Stücklisten und Montageanleitungen im Zusammenhang mit dem Taylorismus erkannt. In eigenen zentralen Organisationseinheiten werden diese Dokumente für alle zu fertigenden Produkte entwickelt und verwaltet. Über die zentralen Stücklisten- und Arbeitsplanverwaltungen werden wesentliche Beiträge zur Wertschöpfung der Unternehmen erbracht. Sie verhindern z.B., daß gleiche Teile

unter unterschiedlichen Namen zu teuer beschafft werden. Die wichtigsten Arbeitsergebnisse dieser Verwaltungen sind: Stücklisten, Struktur- oder Baugruppenstücklisten, Teileverwendungslisten, Montage- bzw. Arbeitsanweisungen sowie Bezugsquellennachweise. Die Stücklistenverwaltung liefert nicht nur Informationen für die eigentliche Fertigung sondern auch die Basisdaten für viele andere Bereiche in den Unternehmen z.B. Arbeitsvorbereitung, Materialwirtschaft, Einkauf, Kundenbetreuung, Ersatzteilhaltung, Produktwartung etc.

Ein gutes Beispiel für die Bedeutung der Stücklistenorganisation sind die Rückrufaktionen in der Autoindustrie. Wenn sich zeigt, daß in der Vergangenheit ein unzuverlässiges Teil verwendet wurde, so kann über die Teileverwendungsnachweise ermittelt werden, in welchen Baugruppen und Modellen dieses Teil verwendet wurde. Analog wäre das Problem des Jahrhundertwechsel für die Softwareentwickler kein so großes gewesen, wenn eine Dokumentation gezeigt hätte, in welchen Programmen und Unterprogrammen mit einer Jahresangabe gearbeitet wurde.

Die Begriffe und Methoden der Fertigung lassen sich auf die Softwareentwicklung übertragen. Die nachfolgende Tabelle zeigt die Parallelität in den Resultaten der Fertigung und der Softwareproduktion:

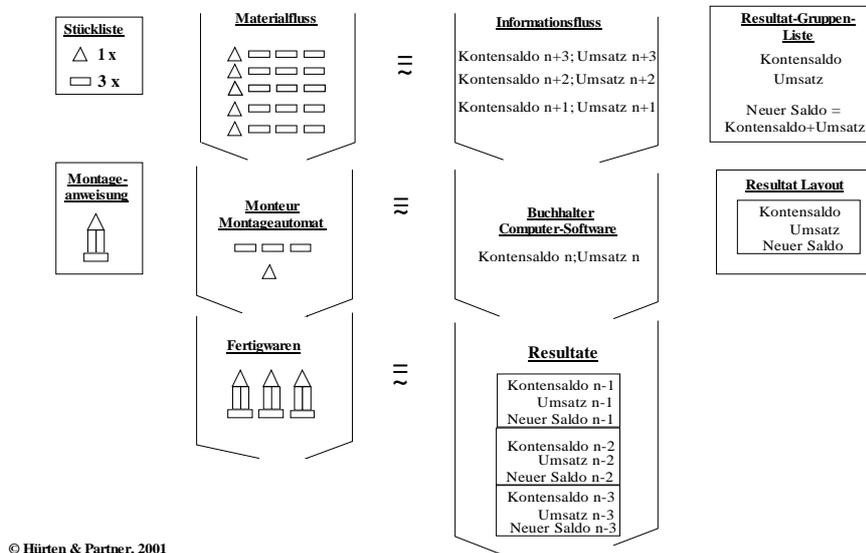
Von Anderen lernen.

Fertigung	Softwareprodukt
Die Ergebnisse der Fertigung sind Baugruppen	Die Ergebnisse der Softwareproduktion sind Resultatgruppen (Informationsgruppen).
Baugruppen können nach unterschiedlichen Montageanleitungen und auf unterschiedlichen Maschinen erstellt werden	Resultatgruppen können in unterschiedlichen Layouts und über unterschiedliche Medien zur Verfügung gestellt werden.
Jede Baugruppe besteht aus Teilen.	Jede Resultatgruppe besteht aus Informationselementen.
Jedes Teil hat mindestens einen Lagerort bzw. Lieferanten	Jedes Informationselement hat mindestens ein Quellelement aus einer Datei bzw. Schnittstelle.
Jedes Teil selbst kann eine Baugruppe sein, die wiederum ihre Stückliste und Montageanweisung hat.	Ein Informationselement kann aus mehreren, nach Regeln verknüpften Quellelementen entstehen.

Das nachfolgende Bild zeigt die Parallelen zwischen der Planung und Fertigung von materiellen Produkten und Software.

Von Anderen lernen.

Analogie Fertigung und Softwareentwicklung



09.02.2006

© R. Hürten; Funktionales Informationsmodell

5

Wenn ein Industrieprodukt gefertigt werden soll, dann muß die Fertigungsstelle wissen, welche Einzelteile oder Baugruppen wie, in welcher Reihenfolge, auf welchen Maschinen zusammenzubauen sind. (Stückliste und Montageanweisung)

Wenn ein Softwareresultat erzeugt werden soll, so muß der Softwareersteller wissen, welche Informationen nach welchen Regeln, in welcher Anordnung und über welches Medium den Anwender geliefert werden sollen. (Resultatinhalt und –layout).

Die Methodik der Beschreibung des Produktes mit Material und Montage und des Softwareproduktes mit Resultatinhalt und Layout ist in beiden Fällen gleich.

Wenn offensichtlich zwischen der industriellen Fertigung und der Softwareentwicklung in Begriffen und Methoden Analogien bestehen, dann werden sicherlich auch die Erfahrungen der Fertigung in der Softwareentwicklung zu nutzen sein.

Dieser Aufsatz wird zeigen, daß in der Softwareentwicklung funktionale Informationsmodelle in Form von Resultat-Gruppen die gleiche zentrale Stellung einnehmen werden wie die Stücklisten in der Fertigung.

3 Das Funktionale Informationsmodell

Was bewirkt Software?

- Mit Hilfe von Software stellen wir
 - in einer virtuellen Welt
 - reale oder als real angenommene (simulierte)
- Zustände,
- Vorgänge und
- Regeln

dar.

Bei dem Begriff Software sind wir heute allzu sehr auf den technischen Vorgang der Programmierung fixiert. Wenn die Frage gestellt wird, was ist Software, so werden wir durchweg die Antwort erhalten: „Software ist eine Folge von Befehlen, die von Computern ausgeführt werden können, also eine Arbeitsanweisung an den Computer.“ Diese Antwort orientiert sich daran, wie sich Software darstellt und nicht was sie darstellt.

Was Software darstellt, sind Abläufe aus der realen Welt. So stellt z.B. ein Materialkonto die Mengenbewegungen eines Artikels dar. Die reale Materialbewegung wurde in der manuellen Buchhaltung auf einem Kontenblatt dargestellt. Damit dieser Vorgang einheitlich dargestellt wurde, erstellte die Organisationsabteilung etwa folgende Arbeitsanweisungen für die Materialbuchhalter: „Materialentnahmen werden wie folgt bearbeitet: Es ist die Materialkontokarte für den Artikel zu ziehen, der auf dem Materialentnahmeschein steht. Übertrage Datum, Menge, Kennzeichen des Entnehmers in die erste freie Buchungszeile ein und errechne den neuen Saldo in der letzten Spalte. Wenn der Bestand gegen Null geht, setze einen roten Reiter auf die Karte und stelle sie zurück in den Kontentrog.“ Die Arbeitsanweisung informierte den Mitarbeiter, wie er einen realen Vorgang auf der Kontenkarte zu simulieren hatte.

Unter diesem Gesichtspunkt ist die Software nichts generell Neues. Es hat sich lediglich der Adressat der Anweisung geändert.

In diesem Buch werden wir die Software vorwiegend unter dem Aspekt des funktionalen Inhaltes betrachten. Wir werden ihre Bedeutung wie folgt definieren:

Mit Hilfe von Software stellen wir reale oder als real angenommene (simulierte) Zustände, Vorgänge und Regeln mit Hilfe von Computern in einer virtuellen Welt dar.

Wie in der Mathematik muß man bei der Beschreibung der Zustände, Vorgänge und Regeln zwischen der algebraischen und der numerischen Ebene unterscheiden. Das funktionale Informationsmodell entspricht der algebraischen Ebene.

Zustände sind Beschreibungen von realen Personen, Gegenstände, Rechten, Ansprüchen oder Situationen und ihren Eigenschaften. Zustände sind mit Momentaufnahmen zu vergleichen. Sie werden von der Software in Form von Resultaten dargestellt, bzw. in den Computern in Form von Dateien und Tabellen beschrieben und vorgehalten/gespeichert.

Reale Zustände

- Zustände sind Beschreibungen von realen
 - Personen
 - Gegenständen
 - Rechten
 - Ansprüchen

- Zustände werden in Ausgaben dargestellt oder in Form von Dateien oder Tabellen beschrieben und vorgehalten.

Reale Zustände

- Beispiel: Reale Person.
Kunde wird beschrieben u.a. durch
 - Name
 - Anschrift
 - Kommunikations-Verbindungen
 - Geschlecht
 - Alter
 - Getätigter Umsatz

Reale Zustände

- Beispiel: Gegenstand.
Maschine wird beschrieben u.a. durch
 - Bezeichnung
 - Nummer
 - Maschinengruppe
 - Standort
 - Lieferant
 - Aktuelle Drehzahl
 - Aktuelle Temperatur
 - Kritische Drehzahl
 - Kritische Temperatur

Reale Zustände

- Beispiel: Abstrakter Anspruch.
Debitorenkonto wird beschrieben u.a. durch
 - Konto-Nummer
 - Kunde
 - Datum letzter Buchung
 - Saldo
 - Umsatz Soll
 - Umsatz Haben
 - Kreditlimit
 - Bonitätskennzeichen

Vorgänge beschreiben die Transformation eines Zustandes „X“ in einen Zustand „Y“. Die Vorgänge werden durch ein Ereignis (Auslöser/Trigger) ausgelöst. Die Transformationen erfolgen nach vorgegebenen Regeln.

Reale Vorgänge und ihre Regeln

Beispiel: Kauf / Verkauf

Hier sind u.a. folgende Zustände zu beachten:

- Warenbestand bei Käufer
- Geldbestand bei Käufer
- Warenbestand bei Verkäufer
- Geldbestand bei Verkäufer

Trigger ist eine Bestellung oder Auftrag

Die Zustände verändern sich durch Kauf/Verkauf nach folgenden **Regeln**:

Warenbestand bei Käufer	+ Liefermenge
Geldbestand bei Käufer	- Kaufpreis
Warenbestand bei Verkäufer	- Liefermenge
Geldbestand bei Verkäufer	+ Kaufpreis

Reale Vorgänge und ihre Regeln

Beispiel: Kauf / Verkauf

Hier sind u.a. folgende Zustände zu beachten:

- Warenbestand bei Käufer
- Geldbestand bei Käufer
- Warenbestand bei Verkäufer
- Geldbestand bei Verkäufer

Trigger ist eine Bestellung oder Auftrag

Die Zustände verändern sich durch Kauf/Verkauf nach folgenden **Regeln**:

Warenbestand bei Käufer	+ Liefermenge
Geldbestand bei Käufer	- Kaufpreis
Warenbestand bei Verkäufer	- Liefermenge
Geldbestand bei Verkäufer	+ Kaufpreis

Die Beispiele zeigen, daß es für die Beschreibung der Zustände, Vorgänge und Regeln keine Rolle spielt, für welchen Anwendungsbereich eine Software geschrieben wird. Die Beschreibung befindet sich immer auf der gleichen Abstraktionsebene. In der Praxis dagegen wird jedoch u.a. unterschieden zwischen kommerzieller, technisch-wissenschaftlicher, embedded und ähnlicher Software. Für die funktionale Beschreibung kann diese Unterscheidung nicht aufrecht erhalten werden.

An dieser Stelle muß nochmals auf eine besondere Situation im Bereich der Softwareentwicklung hingewiesen werden. Es wird zu wenig beachtet, daß die Softwareentwicklung auf zwei verschiedenen Ebenen abläuft und die Aktivitäten auf diesen Ebenen unterschiedliches Wissen und Erfahrungen verlangen.

Die erste Aktivität hat die Realität zu beschreiben, die mit Hilfe der Software virtuell dargestellt werden soll. Hier wird die Fähigkeit gefordert, die realen Zustände, Vorgänge und deren Regeln genau zu kennen und im Detail exakt zu beschreiben. Das war früher die Aufgabe der Organisatoren. Diese hatten die realen Vorgänge in Organisationsanweisungen so zu beschreiben, daß der normale Mitarbeiter danach seine Arbeiten richtig und optimal erledigen konnte.

Die „Organisationsanweisungen“ an den Computer müssen im Vergleich zu denen an den Menschen absolut lückenlos und eindeutig sein. Fehler erkennen und beheben kann der Computer nur dann, wenn ihm wiederum lückenlos und eindeutig bereits beschrieben wurde, woran der Fehler erkannt werden kann und welche Anweisungen er beim Erkennen des Fehlers ausführen muß.

Für die Beschreibung der realen Vorgänge sind auch heute noch keine speziellen IT-technische Kenntnisse erforderlich. Wichtig ist die Kenntnis der Vorgänge, ein Gespür für Zusammenhänge und die Fähigkeit exakt zu formulieren.

Die zweite Aktivität in der Softwareentwicklung gleicht der eines Übersetzers. Da der Computer im Gegensatz zum normalen Mitarbeiter die zwischenmenschlichen Anweisungen nicht verstehen kann, sind die Ergebnisse der ersten Aktivität, die Funktionsbeschreibung, in den Maschinencode der Computer zu übersetzen. Erst auf der Grundlage solcher Programmcodes kann der Computer die vom Anwender verlangten Resultate richtig und zeitgerecht liefern.

Diese Übersetzer, sprich die Programmierer, benötigen die Kenntnis der Programmiersprachen und ein Gespür für die Zusammenhänge in und zwischen der Soft- und Hardware.

Die vorliegende Ausarbeitung beschäftigt sich ausschließlich mit der ersten Aktivität, der Beschreibung der Realität.

4 Funktionsbeschreibung

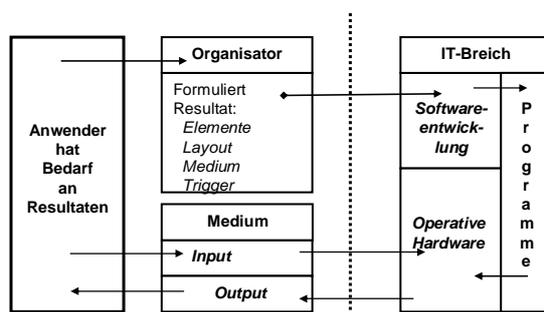
Bevor wir weiterhin von der Funktionalität der Softwaresprechen müssen wir eine Definition dieses Begriffes geben.

Die Funktionalität einer Software

- Die Funktionalität einer Software ist die Summe aller in ihr enthaltenen Abbildungen von realen Zuständen und Vorgängen mit deren Regeln.
- Die Software liefert den Anwendern die Abbilder der Realität in ihren Resultaten.
- Demnach ist für den Anwender die Funktionalität einer Software identisch mit der Summe aller Resultate, die er erhält.
 - Resultate können über beliebige Medien bereitgestellt werden.

Die Funktionalität der Software definiert sich demnach über die in ihr abgebildete Realität und den Resultaten, in denen diese dargestellt werden.

Abbild erstellen – Resultate erzeugen



[return](#)

Aus der vorstehenden schematischen Darstellung der Verbindungen zwischen Anwender und IT-Bereich erkennt man, daß zwei wichtige Verknüpfungen existieren: die zwischen dem Anwender und dem Organisator sowie die zwischen dem Organisator und dem IT-Bereich. Diese Verknüpfungen sind die Grundlage für die Funktionalität der Software.

C:_schriftwechsel\Lotsuite\wordpro\ppbuch.doc Erstelldatum 21.07.2006 12:50

gen zwischen diesen Bereichen existieren. Die liegen auf den planerischen und operativen Ebenen.

Die Darstellung zeigt aber auch die Verantwortung des Anwenders, für die Richtigkeit und Vollständigkeit der Resultate, die er über die Software geliefert bekommt. Der IT-Bereich hat nur die Fehler zu verantworten, die durch eine falsche Codierung oder durch Hardwarefehler verursacht werden.

Sicherlich ist der normale Sachbearbeiter auf der Seite der Anwender nicht in der Lage die Zusammenhänge der Arbeitsabläufe richtig zu erkennen. Das Gleiche gilt aber auch für die IT-Spezialisten, die ihre Ausbildung vorwiegend in der Methodik und Technik der Soft- und Hardware hatten und weniger in den speziellen Anwendungen.

Es wäre deswegen sinnvoll, wieder eine Stelle einzuschalten, die Anforderungen der Anwender kennt und in der Lage ist, diese zu beschreiben. Das Berufsbild des Organisators war darauf ausgerichtet, Arbeitsabläufe zu gestalten und zu beschreiben.

Funktionsanforderungen

- Damit der Softwareentwickler ein Programm erstellen kann, muß der Anwender folgende Anforderungen liefern:
 - Welche Resultate mit welchen Informationen benötigt werden.
 - Über welche Medien die Resultate zu liefern sind.
 - Wie die Informationen in den Resultaten anzuordnen sind. (Layout)
 - Zu welchen Terminen und unter welchen Bedingungen die Resultate bereitzustellen sind. (Trigger)

Ein wichtiger Aspekt bei den vorstehenden Anforderungen ist, daß das Medium der einzige Bezug zu der erforderlichen Arbeitstechnik ist. Dabei werden jedoch keine Vorschriften gemacht, wie die Technik das Medium zu bedienen hat. Daß dies so ist, zeigt folgendes Beispiel: Auch in einer manuellen Organisation muß dem Sachbearbeiter vorgegeben werden, ob er sein Arbeitsergebnis mündlich, in einer Aktenotiz, Brief, Fax etc. an den Adressaten weiterzugeben hat. Welche Hilfsmittel er dabei nutzt, hängt dabei von der technischen Ausstattung seines Arbeitsplatzes ab. Als vor Jahrzehnten vom Federhalter zur mechanischen Schreibmaschine und dann zur elektrischen Speicherschreibmaschine gewechselt wurde, mußten nur in den Arbeitsanweisungen nur das Medium und in Abhängigkeit davon eventuell das Layout geändert werden.

4.1 Resultatgruppen

Resultat-Gruppen (RG)

- In einer Resultat-Gruppe werden alle Informationen einer Funktionsanforderung zusammen gefaßt, die ein genau bestimmtes Resultat betreffen:
 - Resultat-Gruppen Identifikation
 - Resultat-Gruppen Elemente
 - Resultat-Gruppen Medium
 - Resultat-Gruppen Layout
 - Resultat-Gruppen Trigger

Alle Ergebnisse einer Software, die die Computer liefern, sind Informationen, die in Gruppen zusammengefaßt, in einer bestimmten Struktur über ein Medium dem Anwender zu einem geforderten Zeitpunkt zur Verfügung gestellt werden. Diese Gruppen von Informationen bezeichnen wir im Folgenden als Resultat-Gruppen (RG).

Adressaten (Anwender) der Resultat-Gruppen können sowohl natürliche Personen als auch technische Geräte, Schnittstellen, Telephonleitungen etc. aber auch Datenbestände sein. Die Resultate können in unterschiedlichen Layouts über unterschiedliche Medien angeboten werden.

Unter dem Layout versteht man die Struktur und Reihenfolge, in der die Informationen der Resultatgruppen dem Adressaten zur Verfügung gestellt werden.

Der Trigger oder Auslöser definiert eine Resultatgruppe in einem Workflow-

Resultat-Gruppen dienen der Beschreibung der Funktionalität einer Software, so wie sie sich dem Anwender an der Schnittstelle Anwender/Computer darstellt. Sie haben außer der Angabe des Mediums keinen direkten Bezug zu den Computer- und Software-Systemen, die zur Gewinnung der Ergebnisse genutzt werden

- Bedeutung und Aufbau der Resultat-Gruppen

Die Resultat-Gruppen sind die langlebigsten Dokumentationen zu einer Software. Bis auf die Angaben zum Medium beschreiben sie ausschließlich die Funktionalität der Software, d.h. sie beschreiben die Realität, die in der Soft-

ware abgebildet ist. Sie sind vollkommen unabhängig von den Hard- und Softwaresystemen. Sie werden z.B. bei rein technischen und methodischen Migrationen einer Software keinen Änderungen unterworfen. Der Inhalt und die Darstellung der Resultatgruppen werden allein von der Anwendung bzw. dem Anwender bestimmt. Schon unter diesem Aspekt der Stabilität ist es sinnvoll, den Resultat-Gruppen besondere Bedeutung beizumessen. So bleibt z.B. der informative Inhalt der Resultat-Gruppe „Saldenliste“ gleich, unabhängig davon, ob die Rechnung dem Anwender über das Medium Drucker, Bildschirm oder Internet zur Verfügung gestellt wird. Resultatgruppen sind kein spezielles IT-Organisationsmittel. So könnte z.B. die Beschreibung einer Resultatgruppe „Saldenliste“ bei einer manuellen Kontenbuchhaltung wie folgt lauten: Übertragen Sie bitte jeden Ultimo folgende Werte aus den Kontokarten auf die Saldenliste mit folgendem Spaltenaufbau: Kontonummer, Name Kontoinhaber, Datum letzte Buchung, Soll/Haben saldo und bilden Sie die Summe aller Salden getrennt nach Soll und Haben.

Betrachtet man die Struktur und den Inhalt der Resultargruppen, so wird man fünf Ebenen finden:

- Resultat-Gruppen-[Identifikation](#)
- Resultat-Gruppen-Elemente
- Resultat-Gruppen-[Layout](#)
- Resultat-Gruppen-[Medium](#)
- Resultat-Gruppen-[Auslöser](#)

4.1.1 Resultat-Gruppen Identifikation

Resultat-Gruppen-Identifikation

- Resultat-Gruppen Identifikation dient nicht nur der Identifikation, sondern auch zur Beschreibung ihrer Charakteristika.

Name der Resultat-Gruppe	Rechnung
Ablage-Ort	C:\Ordner\Rechnung
Erstellungsdatum	02.02.06
Versionsnummer	2.1
Org.-Verantwortlicher	H. Mustermann
Nutzer	Vertrieb
Datum letzt. Änderung	12.02.06
Resultat-Gruppen-Tabelle	Rechnung
Medium	Drucker
Layout	Rechnung
Trigger	Auslieferung der Ware oder Termin für Sammelrechnung

Die Resultat-Gruppen-Identifikation (RGI) dient der Verwaltung und Katalogisierung der Resultat-Gruppen.

Im Hinblick auf die Funktionalität gibt sie nur eine schlagwortartige Inhaltsangabe. Sie benennt mit einem Schlagwort logisch zusammengehörende Resultatfelder, Über den genauen Inhalt einer Software gibt sie keine detaillierten Informationen.

Die Result-Gruppe-Identifikation ist unter zwei Aspekten zu betrachten.

Der wichtigste Aspekt ist die Benennung und Archivierung. Dies heißt, über einen Namen ist die Resultgruppe zu identifizieren und wieder verfügbar zu machen.

Der zweite Aspekt ist, die Historie in Bezug auf Veränderungen, Nutzung und Verantwortung festzuhalten. Unter diesem Gesichtspunkt werden z.B. folgende Informationen in die Identifikation

Die wichtigste und immer vorhandene Information in der Identifikation ist der Name/die Bezeichnung der Resultat-Gruppe. Sie beschreibt in der Regel die funktionale Bedeutung der Resultat-Gruppen.

Alle zusätzliche Informationen, wie z.B. Erstellungsdatum, Datum und Art der Änderungen, Gültigkeitsbereich, Verantwortlicher für den Inhalt, Nutzer der Resultat-Gruppe u.ä. werden durch die Gestaltung der Resultat-Gruppe-Organisation im Unternehmen bestimmt.

Die Identifikationen der Resultat-Gruppen enthalten sinnvollerweise nicht nur Angaben zu ihrem Aufbewahrungsort. Sie verweisen ferner auf die für sie organisatorisch Verantwortlichen und auf die zu der Resultat-Gruppe gehörenden Tabellen, Layouts und Medien.

Die Resultat-Gruppen-Identifikationen sind eine wichtige Informationsbasis über den gesamten Verlauf des Software Life Cycles. Man kann sie als Anker für eine resultgruppe bezeichnen.

Die Resultat-Gruppen-Identifikation ist Mindestbestandteil jeder Anwenderanforderung. Dies soll heißen, daß zu Mindest eine Aufzählung aller Resultat-Gruppen vorhanden sein muß.

4.1.2 Resultat-Gruppen-Element

Resultat-Gruppen-Elemente (RGE) sind Informationen, die den Inhalt und damit die Bedeutung des Resultats beschreiben. Jede Resultat-Gruppe hat mindestens ein Resultat-Gruppen-Element.

Jedes Resultat-Gruppen-Element hat mindestens ein Quellelement, aus dem es gebildet wird. Ein Resultat-Gruppen-Element kann auch aus mehreren Quellelementen nach vorgegebenen Regeln gebildet werden.

Beispiel:

Resultat-Gruppen-Element	Quellelement
Kundennummer in Rechnung	Kundennummer in der Eingabe
Zeilenpreis in Rechnung	(Liefermenge im Lieferschein) X (Grundpreis im Artikelsatz)

Resultat-Gruppen-Element

- Resultat-Gruppen-Elemente sind Informationen, die den Inhalt des Resultats beschreiben.
- Jede Resultatgruppe hat mindestens ein Element.
 - **Resultat-Elemente: Rechnung**
 - Anschrift
 - Rechnungsnummer
 - Kunden-Nummer
 - Betreff
 - Artikelnummer
 - Artikelbezeichnung
 - Bestellmenge
 - Liefermenge
 - ...

09.03.2006

© R. Hürten; Funktionales
Informationsmodell

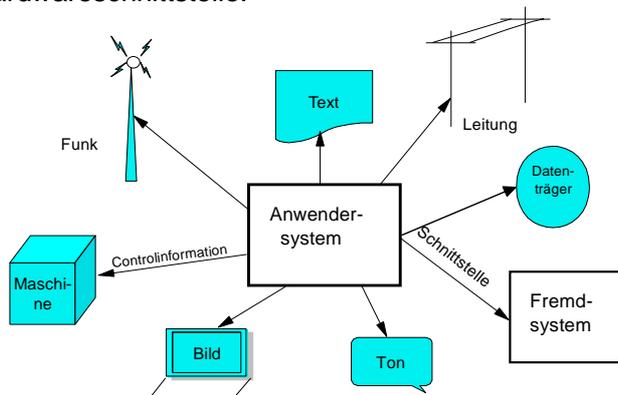
19

Mit jedem Resultat-Gruppen-Element wächst die Größe der Resultat-Gruppe um eine Einheit. Die Anzahl der Resultat-Gruppen-Elemente bestimmt die Größe der Resultat-Gruppe. Auf der Basis von Resultat-Gruppen-Elemente können Resultat-Gruppen im Hinblick auf ihre Größe verglichen werden. Über die Resultat-Gruppen-Elemente wird definiert, was ein Resultat liefert.

4.1.3 Resultat-Gruppen-Medium

Resultat-Gruppen-Medium

- Mit dem Resultat-Gruppen-Medium bestimmt der Anwender die Hardwareschnittstelle.



09.03.2006

© R. Hürten; Funktionales
Informationsmodell

20

Die Medien, über die Resultat-Gruppen dem Anwender angeboten werden, sind unter zwei Aspekten von Bedeutung.

Als Erstes spiegelt sich in dem Medium das Organisationskonzept des Anwenders wieder. Der Anwender bestimmt aus der Sicht seiner Anwendung, über welches Medium das Resultat geliefert werden soll. Zweitens bestimmt das Medium die Realisierung im Hinblick auf die technischen und die methodischen Lösungen.

Das Medium ist eine technische Beschreibung der Resultat-Gruppe und keine funktionale. Das Medium bedarf zu seiner vollständigen Beschreibung keiner Tabellen oder Grafiken wie die Elemente und das Layout. Über das Medium wird definiert, wo ein Resultat bereitzustellen ist.

4.1.4 Resultat-Gruppen-Layout

Resultat-Gruppe-Layout

- Das Resultat-Gruppen-Layout bestimmt die Anordnung der Resultat-Gruppen-Elemente bei der Übergabe an das Medium.

Positionsnummer	Element	Zeilen-Nr.	Position	Länge
1	Datum	1	40	8
2	Anschrift	4-10	4	25
3	Kundennummer	12	4	10
4	Rechnungsnummer	12	40	8

Das Layout einer Resultat-Gruppe bestimmt, in welcher Reihenfolge und Anordnung die Elemente einer Resultat-Gruppe über ein Medium dargestellt werden. Werden gleiche Resultat-Gruppen-Elemente in unterschiedlichen Layouts dargestellt, so bildet jedes Layout eine neue Resultat-Gruppe. Die Layouts können unterschiedlich dargestellt werden. Die gängigsten Darstellungen sind Formularemuster, Bildschirmlayouts, Tabellen und Strings.

Es ist zu beachten, daß durch ein Layout nicht zwangsweise das Medium bestimmt wird. So kann z.B. das gleiche Layout für eine Rechnung auf einem Bildschirm angezeigt, über einen Drucker ausgegeben oder per Email versandt werden.

Sowohl durch die Resultat-Gruppen-Elemente als auch durch das Resultat-Gruppen-Layout werden nicht automatisch geben keine Hinweise und/oder Anforderungen an die spätere technische oder methodische Realisierung der Software definiert.

Über das Layout wird definiert wie ein Resultat zu liefern ist.

4.1.5 Resultat-Gruppen-Auslöser

Resultat-Gruppen-Trigger

- Der Resultat-Gruppen-Trigger definiert wann und unter welchen Bedingungen ein Resultat erstellt werden soll.
 - Manuell: Das Resultat kann von dem Anwender nach Bedarf angefordert werden
 - Zeitlich: Das Resultat wird durch das Erreichen eines bestimmten Datums ausgelöst
 - Workflow gesteuert: Das Resultat wird durch das Erreichen einer vorgegebenen Position in einem Workflow gestartet.
 - Ergebnis gesteuert: In Abhängigkeit eines bestimmten Ergebnisses in einem Resultat wird eine neue Resultatgruppe aufgerufen.

In dem Resultat-Gruppen-Auslöser wird festgelegt, wann und unter welchen Bedingungen das Resultat erstellt werden soll.

Resultat-Gruppen-Auslöser können durch unterschiedliche Ereignisse aktiviert werden.

- Manuell: Das Resultat kann vom Anwender nach Bedarf angefordert werden.
- Zeitlich terminiert: Das Resultat wird durch das Erreichen eines vorgegebenen Datums gestartet.
- Workflow gesteuert: Das Resultat wird durch Erreichen einer vorgegebenen Position in einem Workflow gestartet.
- Resultat gesteuert: In Abhängigkeit eines bestimmten Ergebnisses wird aus dem aktuellen Resultat ein weiteres aufgerufen.

Die Ereignisse selbst können durch zusätzliche Bedingungen beeinflusst werden, z.B. Prüfung auf Berechtigung des Anwenders.

Mit dem Resultat-Gruppen-Auslöser wird nicht eine Funktionalität beschrieben, sondern deren Eingliederung in einen Arbeitsablauf. Sie beschreibt nicht das WAS einer Funktionalität, sondern das Wann.

4.1.6 Resultat-Gruppen-Zusammenstellung

Die Resultat-Gruppen-Zusammenstellung (RGZ) liefert ein Verzeichnis aller Resultat-Gruppen einer Software.

In der Resultat-Gruppen-Zusammenstellung werden alle Resultat-Gruppen mindestens mit ihren Namen, ihrem Layout, ihrem Medium und ihrem Auslöser aufgeführt.

Die Resultat-Gruppen-Zusammenstellung ist eine erste globale Übersicht über die Ergebnisse, die eine Software liefert. Sie zeigt für alle Resultate die Verknüpfungen zwischen Resultat-Gruppen, Layout und Medien.

Resultat-Gruppen-Zusammenstellung

- Die Resultat-Gruppen-Zusammenstellung ist das Verzeichnis aller Resultat-Gruppen einer Anwendung.

Lfd. Nr	Resultat-Gruppen-Name	Layout-Name der RG	Medium der RG	Trigger der RG
1	Lieferschein	Lieferschein	Drucker	Bestellung
2	Rechnung	Rechnung	Internet	Auslieferung
3	Rechnungs - ausgangsbuch	RA-Buch	Bildschirm	Rechnung
4	Umsatzstatistik	U-Statistik	Speicher	RA-Buch

- Die Resultat-Gruppen-Zusammenstellung gibt einen ersten Hinweis über Umfang und Art einer Software.

Aus der Resultat-Gruppen-Zusammenstellung lassen sich erste Rückschlüsse über Umfang und Art der Software ziehen. Der Umfang einer Software wächst mit dem Umfang der Resultat-Gruppen-Zusammenstellung, d.h. mit der Anzahl der Zeilen in ihr. Es ist jedoch zu beachten, daß die Resultat-Gruppen selber sehr unterschiedlich groß sein können. Die Zahl ihrer Elemente kann nicht nur theoretisch zwischen einem und einigen hundert schwanken. Die Anzahl der Zeilen in der Resultat-Gruppen-Zusammenstellung kann somit nur als eine grobe Kennzahl zur Messung der Größe einer Software genutzt werden.

Über die Arten der Medien läßt sich der Charakter der Software erkennen (Dialog, Dateiverwaltung, web-application, Prozeßsteuerung etc.). Es ist zu beachten, daß der Anwender alleine nicht über die Art der Medien bestimmen kann. Hier muß er sich mit seinen Vorstellungen an der aktuellen Situation im Hardwarebereich orientieren.

Die Einträge zum Auslöser sind weniger für die planersiche als für die operative Ebene von Bedeutung.

Aus dem Resultat-Gruppen-Auslöser läßt sich der Arbeitsablauf für die beschriebene Funktionalität aufbauen.

4.2 Resultat-Gruppen Tabellen

Resultat-Gruppen-Tabellen

- Die Elemente einer Resultat-Gruppe werden in einer Resultat-Gruppen-Tabelle zusammengefaßt.
- Je nach Informationsstand und Verwendungszweck kann eine Resultat-Gruppen-Tabelle unterschiedliche Inhalte und Aussagekraft haben.
 - Resultat-Gruppen-Elemente-Tabelle (RGET)
 - Resultat-Gruppen-Quellen-Tabelle (RGQT)
 - Resultat-Gruppen-Funktions-Tabellen (RGFT)
- Die Resultat-Gruppen-Tabellen bilden die Basisdokumentation in softwaretechnischer und betriebswirtschaftlicher Sicht.

Um ein Resultat exakt zu beschreiben, müssen die Elemente bekannt sein, die das Resultat dem Anwender in als Information zur Verfügung stellt.

Die zu einem Resultat gehörenden Informationen werden als Elemente in den Resultat-Gruppen-Tabellen dokumentiert.

Je nach Informationsstand und Verwendungszweck kann eine Resultat-Gruppen-Tabelle unterschiedliche Inhalte und Aussagekraft haben. Es ist sinnvoll folgende Tabellen zu unterscheiden:

- Resultat-Gruppen-Elementen-Tabelle (RGET),
- Resultat-Gruppen-Quellen-Tabelle (RGQT),
- Resultat-Gruppen-Funktions-Tabelle (RGFT).

Der Zeitpunkt der Erstellung dieser Tabellen hängt, von der Verfügbarkeit und der Detaillierung der Dokumentation der Anforderungen bzw. Produkte ab. Im Normalfall ist es so, daß im Verlauf eines Entwicklungsprojektes als erste die RGET und als letzte die RGFT erstellt wird.

Die Tabellen RGET, RGQT und RGFT bauen aufeinander auf. RGET ist eine Untermenge von RGQT und RGFT. RGQT ist wiederum eine Untermenge von RGFT.

4.2.1 Resultat-Gruppen-Elementen-Tabelle RGET

Resultat-Gruppen-Elemente-Tabelle (RGET)

- Die Resultat-Gruppen-Elemente-Tabellen (RGET) zeigt die Informationen, die in der Resultat-Gruppe zur Verfügung gestellt werden.

Lfd. Nummer	Resultat-Medium	Resultat-Gruppen-Namen	Resultat-Gruppen-Elemente

- Die RGET entspricht einer einfachen Stückliste in der industriellen Fertigung

Die Resultat-Gruppen-Element-Tabelle RGET zeigt die Informationen, die in einer Resultat-Gruppe dem Anwender - im weitesten Sinne - zur Verfügung gestellt werden. Ihre Struktur ist einfach. Sie sagt nichts über die Herkunft und die Zusammensetzung der Elemente.

Die RGET entspricht einer einfachen Stückliste in der Fertigung und hat für die Softwareplanung und Softwarepflege die gleiche Bedeutung wie die Stückliste in der Produktplanung und -pflege.

Die RGET sollte zweckmäßiger Weise neben den Informationselementen einen Verweis auf die Dokumentation geben, aus der die Tabelle abgeleitet wurde.

Die Summe aller Resultat-Gruppen-Elemente kann als Größe einer Anwendung definiert werden.

Aus der Anzahl der Elemente über alle RGETs kann ermittelt werden, ob ein Resultat oder eine Anwendung mehr oder weniger Informationen liefert als ein anderes bzw. eine andere. Die Summe der Resultat-Gruppen-Elemente ist für den Umfang der Funktionalität einer Software.

Über die Komplexität der Informationsgewinnung sagen die RGETs jedoch nichts aus. D.h. aus einer RGT kann man nicht erkennen, welcher Aufwand erforderlich ist, um die Resultat-Gruppen-Elemente zu erhalten.

Auf der Grundlage der RGET und der Layout-Tabelle sind Benutzeroberflächen bzw. die Schnittstellen exakt beschrieben. Aus den RGTs kann jedoch nicht erkannt werden, woher und nach welchen Regeln die Oberflächen zu füllen sind.

Beispiel einer Resultat-Gruppen-Element-Tabelle RGET:

<i>LFD. NR.</i>	<i>Dokumenta- tionsort</i>	<i>Resultat-Gruppe-Namen</i>	<i>Resultat-Gruppen-Element</i>
1.1	<i>Requirement</i>	<i>Lieferschein</i>	<i>Lieferdatum</i>
1.2			<i>Lieferschein-Nummer</i>
1.3			<i>Kundenanschrift</i>
1.4			<i>Bestelldatum</i>
1.5			<i>Ihr Zeichen</i>
1.6			<i>Artikel-Nummer</i>
1.7			<i>Bestellmenge</i>
1.8			<i>Liefermenge</i>
1.9			<i>Hinweis</i>
2.1	<i>Requirement</i>	<i>Rechnung</i>	<i>Rechnungsdatum</i>
2.2			<i>Rechnungsnummer</i>
2.3			<i>Kundenanschrift</i>
2.4			<i>Lieferschein-Datum</i>
2.5			<i>Lieferschein-Nummer</i>
2.6			<i>Artikelnummer</i>
2.7			<i>Liefermenge</i>
2.8			<i>Stückpreis</i>
2.9			<i>Zeilenpreis</i>
2.10.....		
3.1			<i>Rechnungswarenwert</i>
3.2			<i>Mehrwertsteuer</i>
3.3			<i>Rechnungsbetrag</i>

Die Trennung des funktionalen Inhaltes (RGET) von der Gestaltung (Layout-Tabelle) ist durchaus sinnvoll.

Wenn zum Beispiel ein Anwender eine Statistik nicht ausgedruckt haben will, sondern als Bildschirmausgabe wünscht, so bleibt die RGET unverändert. Lediglich das Layout ist neu zu gestalten.

Bei der Gestaltung der Dokumentationen sollte auch beachtet werden, zu welchem Zeitpunkt sie erstellt werden sollen bzw. können. Die RGET kann bereits erstellt werden, wenn noch keine genauen Vorstellungen über das Medium und das Layout vorliegen. (Z.B. wenn noch nicht festliegt, ob eine Anschriftszeile 12 oder 15-stellig sein muß) Die RGET beschreiben genau nur das, was die Software zu liefern hat. Sie stellen nur die Funktionalität dar und sind vollkommen unabhängig von der Art der Realisierung.

Dies bedeutet, daß die RGETs bereits in einem frühen Stadium eines Projektes verbindlich erstellt werden können.

Um die Bedeutung der RGETs zu erkennen, muß wie folgendes beachten:

Die RGETs sind eine sehr stabile Dokumentation. D.h. sie unterliegen im Verlauf des lifecycle einer Software nur geringen Änderungen, da sie unabhängig von den Wechseln in Methoden und Technik von Hard- und Software.

Von Bedeutung sind die RGETs aber auch für die Softwarewartung. Werden die RGETs in Datenbanken gespeichert, so erhält man nicht nur Aussagen, darüber, aus welchen Informationen sich ein Resultat zusammensetzt. Man bekommt auch die Frage beantwortet, in welchen Resultaten eine bestimmte Information enthalten ist. Aus den RGETs können Informations-Verwendungsnachweise erstellt werden. Das Problem 2000 wäre nicht so kostspielig gewesen, wenn in den Softwareabteilungen RGETs gepflegt worden wären. Die Abfrage „In welchen Resultaten werden die Information Datum, Jahr, Date etc. verwendet?“ hätte alle die Resultate aufgezeigt, in denen eine Änderung zu vermuten gewesen wäre.

Die Informations-Verwendungsnachweise bieten die gleichen Vorteile wie die Teilverwendungsnachweise in der industriellen Fertigung.

4.2.2 Quellen-Gruppen-Tabellen QGT

Die RGET zeigt nur, welche Informationen von der Software geliefert werden. Sie gibt keine Auskunft darüber, aus welchen Quellen die Ergebnisse aufzubauen sind. Wir erhalten aus den RGETs keinen Hinweis darauf, auf welche Art und Weise die Elemente im Resultat gewonnen werden. Ein Element kann z.B. direkt aus einer einzigen Eingabe gewonnen werden. (Beispiel: Ersterfassung eines Kundennamens über eine PC-Tastatur.) Es kann aber auch aus der Verknüpfung mehrerer Informationen aus Eingaben und/oder Datenbeständen gewonnen werden. (Beispiel: Fakturierter Preis = Liefermenge x Einzelpreis - Kundenrabatt – Artikelrabatt;)

Um zu zeigen, aus welchen Informationen ein Ergebnis aufgebaut wird, werden die Resultat-Gruppen-Elementen-Tabellen um mehrere Spalten erweitert, die auf den Ursprung/die Quelle der RGEs hinweisen.

4.2.2.1 Quellgruppen

Die Stücklisten in der Industrie zählen nicht nur die Teile auf, aus denen sich ein Produkt oder eine Baugruppe zusammen setzt. Für die Fertigungsplanung und –steuerung ist es sinnvoll zu wissen, woher die Teile kommen, die zu verwenden sind. Sind es Lagerteile, Zukaufteile oder kommen Sie aus einer Vorstufenfertigung? Die Stückliste soll aber nicht nur eine Aussage über die Herkunft der Teile machen, sondern auch über die Komplexität der Montage der Teile.

Überträgt man diese Aspekte auf Informationsmodelle, so sind zu jedem RGE anzugeben,

- welche Einzelinformationen werden zu dem RGE verknüpft,
- gehören die Einzelinformationen zu einer logischen Informationsgruppe,
- über welches Medium wird die benötigte Einzelinformation zur Bildung des Resultats zur Verfügung gestellt.

Quell-Element

- Jedes Resultat-Gruppen-Element hat mindestens ein Quell-Element.
- Ein Resultat-Gruppen-Element kann durch Verknüpfung mehrerer Quell-Elemente gebildet werden.
- Jedes Quell-Element ist einer Quell-Gruppe zuzuordnen.
- Jede Quell-Gruppe wird über ein Medium bereitgestellt.

Wir benötigen also zur Beschreibung der RGEs Elemente aus „Quellgruppen (QG)“, die wir als Quell-Gruppen-Elemente bezeichnen. Die QG selber werden über unterschiedliche Medien zur Verfügung gestellt bzw. abgerufen. Für alle Quell-Gruppen, die nicht Elemente aus real-time Eingaben/Erfassungen beschreiben, gilt, daß sie selbst einmal eine Resultatgruppe waren. Beispiel: Die Quellgruppe „Kundestamm“, die bei dem Vorgang Fakturierung benötigt wird, war selbst Resultat-Gruppe bei dem Vorgang „Pflege Kundestamm“.

4.2.2.2 Struktur der Quellen--Gruppen-Tabelle RGQT

Quellen-Gruppen-Tabelle (QGT)

- Quellen-Gruppen-Tabellen erweitern die RGET dadurch, daß jedem Resultat-Element die Quell-Elemente zu geordnet werden, aus denen es gebildet wird.
- Die QGT entspricht einer Stückliste die auch Baugruppen mit deren Zusammensetzung enthält.
- Die QGT läßt sich auch als Informations-Verwendungs-Liste nutzen.
- [Info_Modell_rqt.xls](#)

Lfd. Nummer	Resultat-Medium	Resultat-Gruppen-Namen	Resultat-Gruppen-Element	Quell-Gruppen-Element	Quell-Gruppen-Namen	Quell-Gruppen-Medium
-------------	-----------------	------------------------	--------------------------	-----------------------	---------------------	----------------------

Die QGT stellt die simple Tatsache dar, daß ein Computer nur eine Information ausgeben kann, wenn diese direkt oder indirekt im Computer gespeichert ist oder wenn diese über ein Medium des Computers verfügbar ist. Zur Beschreibung der Funktionalität Rabattgewährung ist deswegen notwendig, zu beschreiben, ob sich der Rabattbetrag aus dem Rabattsatz in der Großkundenrabattstafel, in der Sonderrabattstafel oder in der manuellen Eingabe und dem Standardpreis oder dem Messesonderpreis ergibt.

Analog zu den Resultat-Gruppen können Quell-Gruppen unterschiedlicher Art sein und über unterschiedliche Medien zur Verfügung gestellt werden. Beispiele: Eingabe über Bildschirm, Eingabe über DFÜ, Datenbestand in eigener Anwendung, Datenbestand aus fremden Datenbestand, Meßwert etc.. Die Aufzählung zeigt, daß der Begriff Medium breit anzuwenden ist.

Beispiel: Resultat-Quell-Gruppen-Tabelle QGT

	Resultat-Gruppen-Medium	Resultatgruppen (RG)	RG-Element	Quell-Gruppen-Element	Quell-Gruppe	Quell-Gruppen-Medium
1	printer	Lieferschein	Lieferdatum	Tagesdatum	Systeminfo	System
2	.	.	Lieferscheinnummer	Lfd. Nummer Lieferscheine	Faktura-Stamm	Speicher
3	speicher	Faktura-Stamm	Lfd. Nummer Lieferscheine	Lfd. Nummer Lieferscheine	Faktura-Stamm	Speicher
4	printer	Lieferschein	Kundennummer	Kundennummer	Auftrag	Speicher
5	.	.	Kundenanschrift	Kundenanschrift	Kunden-Stamm	Speicher
6	.	.	Bestelldatum	Bestelldatum	Auftrag	Speicher
7	.	.	Ihr Zeichen	Betreff	Auftrag	Speicher
8	.	.	Artikel-Nummer	Artikel-Nummer	Auftrag	Speicher
9	.	.	.	Artikel-Nummer	Artikel-Stamm	Speicher
10	.	.	Artikelbezeichnung	Artikelbezeichnung	Artikel-Stamm	Speicher
11	.	.	Bestellmenge	Bestellmenge	Auftrag	Speicher
12	.	.	Liefermenge	Bestellmenge	Auftrag	Speicher
13	.	.	.	Lagermenge	Lager-Stamm	Speicher
14	Speicher	Lagerstamm	Lagermenge	Lagermenge	Lagerstamm	Speicher
15	Printer	.	Hinweis	Manueller Eintrag	Tastatur	Eingabe
16	Printer	Rechnung	Rechnungsdatum	Tagesdatum	Systeminfo	System

17	.	.	Rechnungs- Nummer	Lfd. Nummer Rechnung	Faktura- Stamm	Speicher
18	.	.	Kundenanschrift	Kundenanschrift	Lieferschein	Speicher
19	.	.	Lieferschein- Datum	Lieferdatum	Lieferschein	Speicher
20	.	.	Lieferschein- Nummer	Lieferschein- Nummer	Lieferschein	Speicher
21	.	.	Artikelnummer	Artikel-Nummer	Lieferschein	Speicher
22	.	.	Liefermenge	Liefermenge	Lieferschein	Speicher
23	.	.	Stückpreis	Stückpreis	Artikel- Stamm	Speicher
24	.	.	Zeilenpreis	Liefermenge	Rechnung	Speicher
25	.	.	.	Stückpreis	Rechnung	Speicher
26
27	.	.	Rechnungs- warewert	Zeilenpreis	Rechnung	Zwischen- speicher
28	.	.	Mehrwert Steuer	Rechnungs- Warenwert	Rechnung	Zwischen- speicher
29	.	.	.	MwSt-Satz	Faktura- Stamm	Speicher
30	.	.	Rechnungsbetrag	Rechnungs- Warenwert	Rechnung	Zwischen- speicher
31	.	.	.	Mehrwert Steu- er	Rechnung	Zwischen- speicher

Die QGTs sind mehr als eine Dokumentation der Ein- und Ausgabe. Sie stellen nicht nur die Funktionalität dar, sondern sie ermöglichen auch, eine Aussage über die Komplexität der Resultate insgesamt und des einzelnen Elementes insbesondere zu machen. Das Verhältnis der Summe der Quellen-Gruppen- Element zu der Summe der Resultat-Gruppen-Elementen ist eine Kennzahl für die Komplexität der Informationsstrukturen.

Die einfachste Informationsstruktur hat den Wert $1/1=1$. Hierbei handelt es sich eine Funktion wie bei der Schreibmaschine. Jedes Zeichen aus der Quelle wird unverändert an das Resultat übergeben.

In dem vorstehenden Beispiel ergeben sich folgende Werte:

	2	4	22	30	10	2
	Medium	Resultatgruppen (RG)	RG-Element RGE	Quell- Element	Quell- Gruppe	Quell- Gruppen- Medium

Aus diesen Ergebnissen kann abgelesen werden, daß sich im Durchschnitt ein Resultatelement aus 1,3 Quellelemente und eine Resultatgruppe aus 2,5

Quellgruppen gebildet werden. Eine solche Datenstruktur würde man als ein einfache bezeichnen.

Die QGT ist eine zweidimensionale Tabelle, die von zwei Seiten aus interpretiert werden kann. Ist die Tabelle nach den Resultaten sortiert, so bildet sie eine Stückliste. Sie zeigt, aus welchen Quellen ein Resultat gewonnen wird. (S.h. auch [Kapitel 6: Ein-/ Ausgabematrix](#)) Wird QGT nach Quell-Gruppen und deren Elemente sortiert, so kann sie im Sinne eines Teileverwendungsnachweises, wie sie in der Industrie üblich sind, genutzt werden. Sie weist für jedes Quell-Gruppen-Element nach, in welchen Resultat-Gruppen es verwendet wird.

Derartige Informationen sind, wie später noch dargestellt wird, vor allem für die Software-Wartung von großer Bedeutung.

Als Beispiel für die Bedeutung der QGT in der Softwarewartung sei hier wiederum das Millennium-Problem hingewiesen. Über die RGQT hätte man alle Resultat-Gruppen aufgezeigt bekommen, in denen ein Datum für die Erstellung eines Resultat-Elementes notwendig war.

4.2.2.3 Resultat-Gruppen-Funktions-Tabellen RGFT

Resultat-Gruppen-Funktions-Tabellen (RGFT)

- Die QGTs beschreiben nur den Inhalt und die Quellen eines Resultates, es fehlen die Verknüpfungen zwischen Resultat und Quelle
 - Deswegen:
- Die RGQT werden durch die Zuordnung der Formeln für die Verknüpfungen zwischen Element und Quellen zur Resultat-Gruppen-Funktions-Tabelle (RGFT) erweitert

Lfd. Nr.	Resultat-Medium	Resultat-Gruppen-Namen	Resultat-Gruppen-Element	Formel Nr.	Formel	Quell-Gruppen-Element	Quell-Gruppen-Namen	Quell-Gruppen-Medium
----------	-----------------	------------------------	--------------------------	------------	--------	-----------------------	---------------------	----------------------

- [Info_Modell_rgft.xls](#)

Die RGQT zeigt zwar die Elemente und deren Quellelemente auf, jedoch nicht die Regeln, nach denen die Quell-Gruppen-Elemente zu Resultat-Gruppen-Elemente verknüpft werden. Die Verknüpfung zweier Quellgruppen-Elemente kann z.B. eine einfache arithmetische Verknüpfung, aber auch ein komplexe wenn – dann – und –oder- Beziehung sein.

Ergänzt man die Quell-Gruppen-Elemente um die Formeln, nach denen sie zu den Resultat-Gruppen-Elemente verknüpft werden, so erhalten wir eine Tabelle, die wir als Resultat-Gruppen-Funktions-Tabelle (RGFT) bezeichnen werden.

Die RGFT zeigt für jedes RG-Element nach welcher Formel es aus den Quell-Elementen zu bilden ist. (in leeres Formelfeld ist als einfacher move zu interpretieren.) Die Formeln schaffen die logische Verbindung zwischen den Resultat und Quellelementen.

Resultat-Gruppen-Medium	Resultatgruppen (RG)	RG-Element	Formel-Nummer	Formel	Quell-Gruppen-Element	Quell-Gruppe	Quell-Gruppen-Medium
printer	Lieferschein	Lieferdatum	.	.	Tagesdatum	Systeminfo	System
.	.	Lieferscheinnummer	.	.	Lfd. Nummer Lieferscheine	Faktura-Stamm	Speicher
speicher	Faktura-Stamm	Lfd. Nummer Lieferscheine	1	(Lfd. Nummer Lieferscheine)+1	Lfd. Nummer Lieferscheine	Faktura-Stamm	Speicher
printer	Lieferschein	Kundennummer	.	.	Kundennummer	Auftrag	Speicher
.	.	Kundenanschrift	.	.	Kundenanschrift	Kunden-Stamm	Speicher
.	.	Bestelldatum	.	.	Bestelldatum	Auftrag	Speicher
.	.	Ihr Zeichen	.	.	Betreff	Auftrag	Speicher
.	.	Artikel-Nummer	2	Artikel-Nummer muß in Artikel-Stamm vorhanden sein	Artikel-Nummer	Auftrag	Speicher
.	.	.	2	Artikel-Nummer muß in Artikel-Stamm vorhanden sein	Artikel-Nummer	Artikel-Stamm	Speicher
.	.	Artikelbezeichnung	.	.	Artikelbezeichnung	Artikel-Stamm	Speicher
.	.	Bestellmenge	.	.	Bestellmenge	Auftrag	Speicher
.	.	Liefermenge	3	Wenn Lagermenge > Bestellmenge; Liefermenge = Bestellmenge. Wenn Lagermenge < Bestellmenge: Liefermenge = Lagermenge	Bestellmenge	Auftrag	Speicher
.	.	.	3	Wenn Lagermenge > Bestellmenge; Liefermenge = Bestellmenge. Wenn Lagermenge < Bestellmenge: Liefermenge = Lagermenge	Lagermenge	Lager-Stamm	Speicher

Speicher	Lagerstamm	Lagermenge	4	Lagermenge = Lagermenge - Liefermenge	Lagermenge	Lagerstamm	Speicher
Printer	.	Hinweis	.	.	Manueller Eintrag	Tastatur	Eingabe
Printer	Rechnung	Rechnungsdatum	.	.	Tagesdatum	Systeminfo	System
.	.	Rechnungsnummer	5	(Lfd. Nummer Rechnung)+1	Lfd. Nummer Rechnung	Faktura-Stamm	Speicher
.	.	Kundenanschrift	.	.	Kundenanschrift	Lieferschein	Speicher
.	.	Lieferschein-Datum	.	.	Lieferdatum	Lieferschein	Speicher
.	.	Lieferschein-Nummer	.	.	Lieferschein-Nummer	Lieferschein	Speicher
.	.	Artikelnummer	.	.	Artikel-Nummer	Lieferschein	Speicher
.	.	Liefermenge	.	.	Liefermenge	Lieferschein	Speicher
.	.	Stückpreis	.	.	Stückpreis	Artikel-Stamm	Speicher
.	.	Zeilenpreis	6	Zeilenpreis=Liefermenge*Stückpreis	Liefermenge	Rechnung	Speicher
.	.	.	6	Zeilenpreis=Liefermenge*Stückpreis	Stückpreis	Rechnung	Speicher
.
.	.	Rechnungswarenwert	7	Rechnungswarenwert=Summe aller Zeilenpreise	Zeilenpreis	Rechnung	Zwischenspeicher
.	.	Mehrwert Steuer	8	Mehr Steuer=Rechnungswarenwert*MwSt-Satz/100	Rechnungswarenwert	Rechnung	Zwischenspeicher
.	.	.	8	Mehr Steuer=Rechnungswarenwert*MwSt-Satz/101	MwSt-Satz	Faktura-Stamm	Speicher
.	.	Rechnungsbetrag	9	Rechnungsbetrag =Rechnungswarenwert +Mehrwert Steuer	Rechnungswarenwert	Rechnung	Zwischenspeicher
.	.	.	9	Rechnungsbetrag =Rechnungswarenwert +Mehrwert Steuer	Mehrwert Steuer	Rechnung	Zwischenspeicher

Die RGFTs haben einen hohen Aussagewert. Sie beschreiben für jedes Ergebnis der Software, welche Informationen für die Gewinnung des Ergebnisses benötigt werden und wie diese zum Ergebnis hin zu verknüpfen sind. Sie vereinigen in sich die Funktionalität der wichtigsten Dokumentationen der industriellen Fertigungsplanung.

Resultat-Gruppen-Funktions-Tabellen (RGFT)

- Die RGFTs entsprechen in der industriellen Fertigung einer
 - Stückliste
 - Baugruppenliste und
 - Montageanweisung
- RGFTs geben Auskunft über die Komplexität einer Software
- Formeln können auch in Entscheidungstabellen hinterlegt werden.

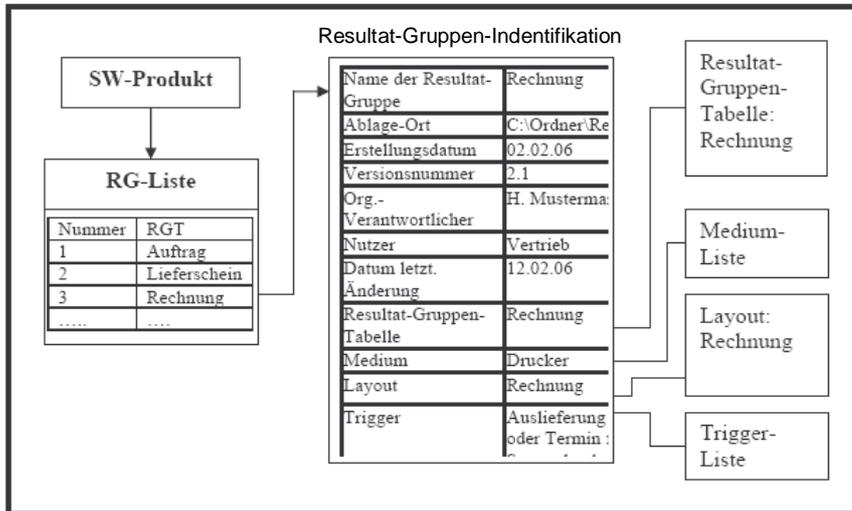
In der RGFT wird die Komplexität der Resultate in Bezug auf Datenstruktur und Verarbeitungslogik voll dargestellt. Auch aus der RGFT lassen sich Kennzahlen für die logische Komplexität eines Resultates ableiten. Das Verhältnis von Anzahl Elemente zu Anzahl Formeln ist eine solche Kennzahl. Die Komplexität der Formeln kann an der Anzahl der Operanden in der Formel gemessen werden.

2	4	22	9	14	20	11	2
Medium	Resultatgruppen (RG)	RG-Element RGE	Formel-Nummer	Formel-Operanden	Quell-Element	Quell-Gruppe	Quell-Gruppen-Medium

Aus den vorstehenden Zahlen für das Beispiel der RGFT lassen sich z.B. zur Bewertung der logischen Komplexität Folgendes sagen:
 Annähernd jedes zweite Resultat-Gruppen-Element wird über eine Formel gesteuert gebildet.
 Die Formeln selber haben eine geringe Komplexität. Sie werden im Schnitt aus weniger als zwei Operanden gebildet.
 Die logische Komplexität wäre als gering einzustufen.

4.3 Funktionales Informationsmodell und relationales Datenmodell

Das Funktionale-Informationsmodell



18.07.2006

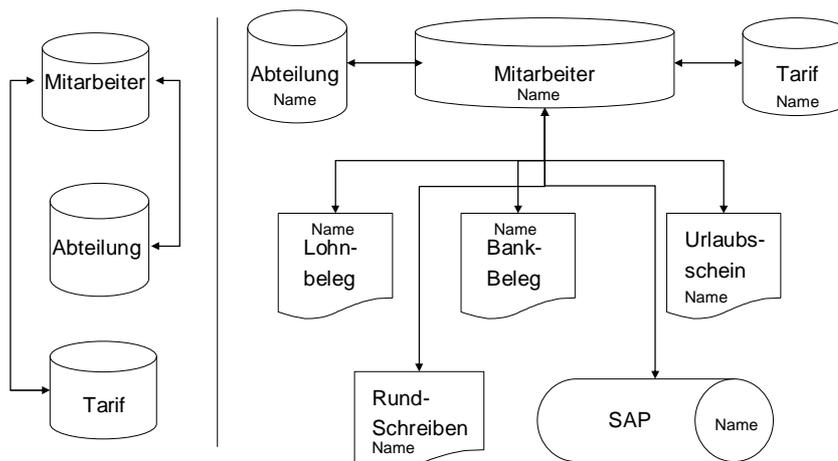
© R. Hurten; Funktionales Informationsmodell

33

Das Zusammenspiel der in den vorstehenden Kapiteln beschriebenen Listen und Tabellen bilden in ihrer Gesamtheit das Funktionale-Informationsmodell für ein Softwareprodukt.

Die relationale Datenbank hat im Vergleich zum Funktionalen-Informationsmodell nur eine begrenzte Aussagekraft.

Datenmodell und Funktionales Informationsmodell



27.05.2006

© R. Hurten; Funktionales Informationsmodell

36

Das relationale Datenmodell beschreibt nur den Zusammenhang einer Untermenge der Medien in einem Funktionalen Informationsmodell. Es stellt nur

die Beziehungen der Speichermedien (Entitäten) untereinander dar. Dabei sind primär nur die Resultat-Gruppen-Elemente (Attribute) von Bedeutung, über die die Verknüpfung der Entitäten erfolgt. Der entscheidende Unterschied zwischen Datenmodell und Informationsmodell ist jedoch, daß das Datenmodell keine Verknüpfung der Entitäten und Attribute zu der Gesamtheit aller im Softwareprodukt genutzter Resultate gibt. Durch die Möglichkeit, die das Funktionale Informationsmodell in dieser Hinsicht bietet, liegt dessen Bedeutung nicht nur in der Dokumentation der Anwenderanforderung. Das Funktionale Informationsmodell bietet die Möglichkeit die Herkunft jeder Information eines Resultates bis zu seiner originären Entstehung zu verfolgen.

4.4 Zusammenstellung der Besonderheiten des Funktionalen Informationsmodells

Besonderheit des Funktionalen Informationsmodell

- Das Funktionale Informationsmodell ist kein speziell auf die IT ausgerichtetes Modell
- Durch die Unabhängigkeit von einer speziellen Technik ist es sehr stabil.
- Durch eine Realisierung in einer Datenbank ergeben sich viele Vorteile für das Softwaremanagement.

4.4.1 Das Funktionale Informationsmodell ist kein speziell auf die IT ausgerichtetes Modell

Aus der Verwendung der Angaben zu dem Medium eines Resultates könnte man schließen, daß es sich hier um eine Information zu einer speziellen IT-Technik handelt. Dies ist jedoch nicht der Fall.

Nehmen wir als Beispiel rein manuell organisierte Arbeitsabläufe. Auch in einer solchen Organisation mußte den Mitarbeiter vorgegeben werden, wie er die von ihm erarbeiteten Resultate darzustellen hatte. Dies konnten zum Beispiel sein: Brief, Karteikarte, Kontokarte, Fax, Telegramm handschriftlich, Schreibmaschine etc.

Das Funktionale Informationsmodell kann zur Beschreibung bzw. Abbildung der unterschiedlichsten Arten von Vorgängen, Regeln und Zustände genutzt

werden, ohne genaue Kenntnis von der organisationstechnischen Realisierung haben zu müssen.

Dies ist eine Eigenschaft, die Verständigungsschwierigkeiten zwischen Anwender und IT-Spezialist erheblich reduzieren kann.

4.4.2 Durch die Unabhängigkeit von einer speziellen Technik ist es sehr stabil

Analysiert man die Projekte in den letzten Jahren, so stellt man fest, daß viele nur die Umstellung bestehender Anwendungen auf eine neue Hard- oder Software war. Dabei wurde die eigentliche Funktionalität kaum verändert. Unveränderte Funktionalität bedeutet aber, daß die Resultat-Gruppen-Tabellen weiterhin genutzt werden können.

Das Funktionale Informationsmodell beschreibt nicht die Aufgaben eines einzelnen Projektes, sondern ein Softwareprodukt über seinen gesamten Lebenszyklus. Die Praxis zeigt, daß in den einzelnen Projekten innerhalb eines Lebenszyklus einer Software immer Teile der Funktionalen Informationsmodellen bearbeitet, verändert bzw. ergänzt werden.

Die Funktionalen Informationsmodelle bilden einen wichtigen Bestandteil der Wissensbasis einer Softwareentwicklung. Auf diese kann immer wieder von Projekt zu Projekt nutzbringend zugegriffen werden.

4.4.3 Durch die Realisierung der Funktionalen Informationsmodelle in einer Datenbank ergeben sich viele Vorteile für das Softwaremanagement.

Aus den Funktionalen Informationsmodellen lassen sich in Zusammenhang mit den Projektberichten wichtige Kennzahlen für das Softwaremanagement ableiten.

Als Beispiele seien an dieser Stelle nur genannt: Produktivitätskennzahlen für einzelne Produkte und Entwicklungsumgebungen, Kalkulationsgrundlagen für Angebote, Definitionen für den Leistungsumfang von Dienstleistungsverträgen u.ä.

Auf die einzelnen Punkt wird in den **Kapiteln** genauer eingegangen.

5 Warum sollte man sich mit dem Funktionale Informationsmodell beschäftigen?

Bevor in dem Kapitel 6 beschrieben wird, wie die Funktionalen Informationsmodelle zu managen sind, soll gezeigt werden, welchen Nutzen ein Unternehmen aus den Funktionalen Informationsmodellen ziehen kann.

5 Warum sollte man sich mit dem Funktionale Informationsmodell beschäftigen?

- Die Einführung und Nutzung der Funktionalen Informationsmodellen verbessert den Softwarebereich in Bezug auf
 - Kosten
 - Termine
 - Qualität
 - Verhältnis von Softwareentwickler zu Anwender
 - Verhältnis von Softwarelieferant zu Softwarekunde
- Dies wird erreicht durch eindeutige, vollständige und verbindliche Anwenderanforderungen und Produktbeschreibungen