

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte bibliografische Daten
sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN 978-3-7281-3761-6 Printausgabe

ISBN 978-3-7281-3762-3 E-Book

Doi 10.3218/3762-3

www.vdf.ethz.ch

verlag@vdf.ethz.ch

© 2016, vdf Hochschulverlag AG an der ETH Zürich

Das Werk einschliesslich aller seiner Teile ist urheberrechtlich geschützt.
Jede Verwertung ausserhalb der engen Grenzen des Urheberrechtsgesetzes ist
ohne Zustimmung des Verlages unzulässig und strafbar. Das gilt besonders
für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung
und Verarbeitung in elektronischen Systemen.

Inhaltsverzeichnis

Vorwort	5
Einleitung	7
Gute Software	7
Verstehen	9
Das Informationsmodell	15
Was ist ein Informationsmodell?	15
Elemente des Informationsmodells	24
Qualität des Informationsmodells	35
Review des Informationsmodells	47
Einsatz des Informationsmodells	53
Arbeiten mit dem Modell	53
Informations- und Prozessmodell	63
Einsatz in verschiedenen Szenarien	68
Erfahrungen mit dem Informationsmodell	75
Widerstände	77
Gewinn	82
Schlusswort	86
Anhang	89
Informationsmodell des Informationsmodells	89
Erweiterungen des Informationsmodells	90
Transformation Informationsmodell – Datenmodell	94
Beispiel eines umgesetzten Informationsmodells	98
Literaturverzeichnis	100
Index	101

Vorwort

Nach der Präsentation eines Informationsmodells war die Reaktion des CEO: «Dafür habt ihr vier Wochen gebraucht. Das ist so klar und offensichtlich, das hätte ich ihnen an einem Nachmittag aufzeichnen können.» Diese Aussage betrachte ich als das grösste Lob, das ich je für meine Arbeit als Informationsmodellierer erhalten habe. Die Mühe, die wir hatten, die Informationen in Erfahrung zu bringen, den Aufwand, den wir betrieben hatten, treffende Namen zu (er)finden, die Diskussionen, die wir geführt hatten, um Unklarheiten und Widersprüchlichkeiten zu beseitigen, all dies war im Endergebnis nicht mehr sichtbar. Wir hatten die Informationswelt dieser Firma – gemäss Aussage des Chefs – klar, treffend und richtig beschrieben. Der Chef verstand die Aussagen auf dem Diagramm.

Darum geht es in diesem Buch. Wie wird gemeinsames Verständnis über alle Ebenen hinweg erreicht? Und wie kann etwas, das verstanden wurde, dokumentiert werden? Welche Dokumentationsform ermöglicht, dass andere möglichst rasch dasselbe Verständnis von der Sache haben? Es geht in diesem Buch nicht um die technischen Belange von Wissen (Speicherung, Daten, Darstellung). Es geht um Inhalte, Essenz und Semantik von Informationen.

Dieses Buch richtet sich an alle, die mit der Verwaltung von Daten und Informationen zu tun haben. Seien es nun IT-Fachleute, Business-Analysten, Personen aus der IT-Organisation oder dem Management oder die Nutzenden aus den Fachabteilungen.

IT-Fachleute erfahren, was der Unterschied zwischen Daten- und Informationsmodellierung ist und welchen Nutzen sie damit in der Kommunikation mit den IT-Laien erzielen können.

Personen aus der IT-Organisation erhalten eine Methode und eine Sprache, mit deren Hilfe sie exakt und verbindlich sowohl mit den IT-Fachleuten als auch mit den künftigen Nutzenden aus ihren Fachabteilungen kommunizieren können.

Business-Analysten erhalten eine Methode und eine Sprache, mit der sie die Ergebnisse ihrer Analyse- und Modellierungsarbeit sehr einfach und für alle Beteiligten verständlich darstellen können.

Personen aus dem Management lernen eine Methode kennen, die es ihnen erlaubt, in kürzester Zeit einen Überblick über die Anforderungen und deren Lösung zu gewinnen. Ohne auf technische Details eingehen zu müssen, erlangen sie Einsicht in die vorgesehene Lösung. Dies erlaubt ihnen, die richtigen Fragen zu stellen und frühzeitig Fehlentwicklungen zu erkennen und zu korrigieren.

Benutzerinnen und Benutzer erhalten die Chance, ihre Beiträge in den umgesetzten Lösungen wiederzufinden und zu verifizieren. Die verständliche Darstellung des eigentlichen Benutzerwissens lässt sie am Projekt teilhaben. Sie können mit den IT-Fachleuten auf Augenhöhe kommunizieren und werden in der künftigen Software ihre Sicht auf die Informationswelt erkennen.

Stefan Berner
Juli 2016

Einleitung

Gute Software

Softwarekrise ist ein Dauerthema, seit es Software gibt. Verschiedene Studien nennen Zahlen von 40–80 % der IT-Projekte, die in den Sand gesetzt wurden. Es werden (schwer überprüfbare) Zahlen von Milliarden von Franken genannt, die mit falsch laufenden Softwareprojekten verloren gehen. Eigenentwicklung von Software ist riskant und meist zu teuer. Der Einsatz von Standardsoftware verursacht häufig mehr Kosten als angenommen, und die Einsparungen bei der Anschaffung werden durch die Mehraufwände bei der Einführung mehr als kompensiert. Softwaresysteme passen nicht zusammen. Schnittstellen sind komplex und fehleranfällig. Viele Beispiele zeigen auf, dass die riesigen Fortschritte der Informatik viel mehr im technischen Bereich (Speicher, Taktraten, Vernetzung) liegen als im inhaltlichen oder qualitativen.

Methoden für die Erstellung von guter Software sind bekannt und erprobt. Warum gibt es so viele Negativbeispiele, die von ausgebildeten Fachleuten nach erprobten Methoden erstellt wurden? Anwenderinnen und Anwender beschreiben mit Unterstützung von Business-Analysten Anforderungen und Konzepte, die ihre Wünsche aus ihrer Sicht korrekt und vollständig wiedergeben. Gut ausgebildete Informatikerinnen und Informatiker schreiben mit modernen Methoden und Werkzeugen Software aufgrund dieser Anforderungen. Trotzdem sind die Kunden unzufrieden. Auch abgesehen von den üblichen Fehlerquellen wie Nachlässigkeit, Unfähigkeit, Schlenndrian, schlechte Arbeitsmoral, schlechte Teamkonstellation etc., passiert es zu häufig, dass gute Leute gute Arbeit leisten und trotzdem ein inakzeptables Ergebnis resultiert.

Unter Softwarequalität versteht man die Gesamtheit der Merkmale und Merkmalswerte eines Softwareprodukts, die sich auf dessen Eignung beziehen, festgelegte oder vorausgesetzte Erfordernisse

Einleitung

zu erfüllen [1]. Auftraggeber empfinden Software demnach als *gut*, wenn sie die Anforderungen in ihrem Sinne erfüllt.

IT-Fachleute kennen ihre Methoden und -Werkzeuge, sie beherrschen ihr Metier. Personen der Fachabteilungen und des Managements wissen, was sie brauchen. Sie wissen um die fachlichen Abläufe, sie haben Wünsche oder Vorstellungen, wie sie gern arbeiten möchten. Umsysteme sind normalerweise bekannt. Die Frage ist nicht, ob dieses Wissen vorhanden ist, sondern wie es in die künftige Software transferiert wird. Nicht fehlendes Wissen führt zu schlechter Software. Es ist der ineffiziente Einsatz des bestehenden Wissens. Es ist die mangelhafte Kommunikation an den Schnittstellen zwischen realer und abstrakter Welt. Verstärkt wird das Problem durch den Umstand, dass alle Beteiligten im guten Glauben handeln, sie hätten sich gegenseitig verstanden.

Motivation für dieses Buch ist die These:

Mangelndes gegenseitiges Verständnis ist die Hauptursache schlechter Software.

Wie kommt es zu Missverständnissen? Warum reden zu viele Leute aneinander vorbei, obwohl sie die gleiche (natürliche) Sprache sprechen? In jeder Umgebung (Firma, Bereich, Sprachregion, Kultur etc.) gibt es Begriffe, die verwendet und von allen verstanden werden. Es ist die täglich benutzte Umgangssprache. Sie ist häufig nicht exakt und wer sie verwendet, der geht davon aus, dass die Empfangenden der Botschaft die verwendeten Begriffe gleich interpretieren wie die Sendenden. Woher sollen Informatikerinnen und Informatiker, die häufig nicht aus derselben Umgebung wie die Auftraggeber stammen, den internen Sprachgebrauch einer Firma kennen? Wie sollen sie die Anforderungsdefinitionen und Wünsche der Kunden verstehen? Wie können sie deren umgebungsspezifische Sprache lernen?

Häufig glauben die Leute, etwas gleich zu verstehen. Sie gehen davon aus, dass andere ähnliche Voraussetzungen haben. Wenn alle Beteiligten einer Kommunikationsrunde behaupten, sie hätten das Gesagte verstanden, ist noch nicht garantiert, dass alle das Gleiche verstanden haben. *Verstehen* hängt von der Sichtweise, dem Hand-

lungszusammenhang, dem Vorwissen, der Umgebung oder kurz vom Kontext¹ ab.

In Softwareprojekten arbeiten häufig Leute zusammen, die nicht das gleiche Kontextwissen haben: externe Berater, externe Programmierende, Lieferanten, Leute aus Management, Fachabteilung und dem IT-Bereich, Leute unterschiedlichster Ausbildungsrichtung und -tiefe.

Für eine unmissverständliche Kommunikation in heterogen zusammengesetzten Gruppen braucht es zwingend einen gemeinsamen Kontext. Dieser Kontext muss in einer Form dokumentiert sein, die von allen Beteiligten verstanden wird. Als (eine) Voraussetzung für gute Kommunikation und damit gute Software braucht es klar und eindeutig definierte Begriffe sowie eine klar und eindeutig definierte Verwendung derselben. Kurz, alle Beteiligten müssen sich verstehen.

Stimmen die Namen und Begriffe nicht, so ist die Sprache konfus.
Ist die Sprache konfus, so entstehen Unordnung und Misserfolg.
Gibt es Unordnung und Misserfolg, so geraten Anstand und gute Sitten in Verfall.

Konfuzius (551–479 v. Chr.)

Verstehen

Erlauben Sie mir, mich mit drei Attributwerten aus unserer Personaldatenbank vorzustellen:

Stefan

Berner

1955

Damit bin ich bereits mitten im Thema dieses Abschnitts: Warum verstehen Sie das? Anders gefragt, hätten Sie

Martin

Peter

8472

auch verstanden? Warum nicht?

Beim ersten Beispiel haben Sie vermutlich Vor- und Nachnamen, aufgrund ihres kulturellen und sprachlichen Wissens, als solche er-

¹ Kontext sei hier umfassend verstanden als Mischung aus Sprache, Kultur, Ausbildung, Erfahrung, Einstellung, Interesse etc.

Einleitung

kannt. Diese Annahme basiert darauf, dass Sie dieses Buch auf deutsch lesen und damit ist es wahrscheinlich, dass Sie deutschsprachige Vornamen wie Stefan, Martin oder Peter als solche erkennen. Der Wert der Zahl sowie der einleitende Hinweis auf die Vorstellung des Autors plus vielleicht ein Foto, das Sie von mir gesehen haben, haben einen Kontext geschaffen. Ich vermute, Sie haben auf einen Jahrgang geschlossen.

Ohne explizites Zusatzwissen können Sie den zweiten Datensatz nicht eindeutig verstehen. Aus der Reihenfolge des ersten Beispiels und der im Deutschen üblichen Reihenfolge Vorname – Nachname kann Martin mit einiger Wahrscheinlichkeit als Vorname interpretiert werden. Sicher ist das aber nicht. Wäre zufällig das zweite Beispiel zuerst aufgeführt gewesen, hätten Sie es vielleicht anders interpretiert. Sie brauchen Struktur- oder Kontextinformation (welches ist Vor-, welches Nachname), um sicher zu sein, dass Sie die Datenwerte richtig, d.h. im Sinne des Autors, der Autorin verstehen.

Die Zahl im zweiten Beispiel ist im aktuellen Kontext offensichtlich kein Jahrgang. Vom Wert her könnte es ein Monatsalär (zumindest in der Schweiz) oder ein Kontostand sein. Tatsächlich ist es eine Schweizer Postleitzahl. In der Schweiz lebende Leserinnen und Leser haben dies möglicherweise erkannt. Damit wird die Vermutung im ersten Beispiel, 1955 sei ein Jahrgang, infrage gestellt. Der Kontext (gleiche Position, gleiche Zifferanzahl) lässt vermuten, dass die beiden Zahlen die gleiche Bedeutung haben. Tatsächlich ist 1955 (nebst meinem Jahrgang) die Postleitzahl von Chamoson im Kanton Wallis.

Gehen wir mit der Interpretation noch einen Schritt weiter. Der Hinweis auf die Personaldatenbank lässt vermuten, dass Vor- und Nachname einen Angestellten bezeichnen. Was zum eindeutigen Verständnis der Aussage fehlt, ist die Bedeutung der Postleitzahl. Was hat ein Ort (das wird in der Schweiz normalerweise mit einer PLZ assoziiert) mit dem Angestellten zu tun? *Wohnt* er dort, *arbeitet* er dort, wurde er dort *geboren*?

Wir können das Kontextwissen, das zum Verstehen der Datenwerte notwendig ist, als Tabelle dokumentieren (siehe Abbildung 1 auf der nächsten Seite). Die Kontextbeschreibung in den beiden ersten Zeilen ist ergänzt um obige Beispieldaten. Eine grafische Dar-

stellung des Kontextwissens allein (ohne Datenwerte) sehen Sie in Abbildung 2).²

Angestellter		wohnt in	Ort
<i>Vorname</i>	<i>Name</i>		<i>Postleitzahl</i>
<i>Martin</i>	<i>Peter</i>		<i>8794</i>
<i>Stefan</i>	<i>Berner</i>		<i>1955</i>

Abbildung 1: Tabellarische Darstellung des Eingangsbeispiels.

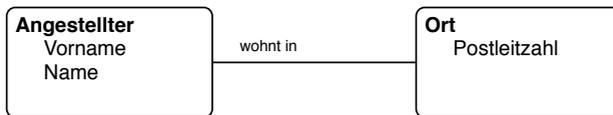


Abbildung 2: Grafische Darstellung des Kontextes des Eingangsbeispiels.

Kehren wir noch einmal zu den Datenwerten zurück.

<i>Martin</i>	<i>Peter</i>	<i>8472</i>
---------------	--------------	-------------

Ohne Kontextwissen können Sie diese drei Werte nicht im Sinne des Autors interpretieren. Daten(-Werte) allein machen keinen Sinn. Sie sind buchstäblich *sinnlos*. Erst die Interpretation durch die Leserin, den Leser erzeugt einen Sinn. Aus den Datenwerten allein kann nie eindeutig herausgelesen werden, was die Werte bedeuten. Die Werte *Martin* und *Peter* werden von den meisten Lesenden als männliche Vornamen erkannt. Dass es auch Nachnamen sein können, zeigt bereits auf, dass Werte allein keinen eindeutigen Sinn ergeben. Nehmen Sie als weiteres Beispiel die folgenden Werte: *Zürich*, *Bern*, *Basel*, *Genf*. Alles klar? Sind es Schweizer Städtenamen? Oder Kantonsnamen? Oder die Namen der Sitzungszimmer einer Firma? Oder die Namen von Lokomotiven der Schweizerischen Bundesbahn?

² Diese Darstellungsform wird im Kapitel *Elemente des Informationsmodells* ab Seite 24 beschrieben.