technische universität dortmund

itpl

Bachelorarbeit

Konzept zur Bereinigung und Anreicherung von Supply-Chain-Daten

Zur Erlangung des akademischen Grades
Bachelor of Science (B.Sc.) Angewandte Informatik

Florian Vielhauer
151178
Dortmund, 27.03.2018

Gutachter:
Prof. Dr.-Ing. Markus Rabe
Prof. Dr. Peter Buchholz

Technische Universität Dortmund
Fakultät für Maschinenbau
IT in Produktion und Logistik

Inhaltsverzeichnis

1. Einleitung

Aufgrund stärker werdender Konkurrenz zwischen Unternehmen, ist eine stetige Überprüfung und Optimierung unternehmensinterner Prozesse ausschlaggebend. Hierzu gehört die Steuerung von Personal- und Finanzverwaltung, ebenso wie die Optimierung von Produktionsprozessen zur Anpassung an veränderliche Faktoren der Ökonomie und Ökologie. Doch gerade die Bedeutung der Logistik hat in den letzten Jahrzehnten immer mehr zugenommen. Ein einwandfreies Funktionieren ist für den Erfolg eines Unternehmens äußerst wichtig geworden. Durch die oftmals dezentrale Aufstellung von modernen Unternehmen, hat die Steuerung von Logistik und Versorgungsvorgängen an Aufwand zugenommen. Meist befinden sich die Produktionsstätten an unterschiedlichen Standorten und Zulieferer liegen global verteilt. Lieferverzug oder Veränderungen von Einkaufspreisen haben direkten Einfluss auf das zu produzierende Endprodukt und somit auf die Konkurrenzfähigkeit der Unternehmen. Zur Unterstützung begründeter Entscheidungen in komplexen logistischen Systemen existieren unterschiedliche Analysemethoden, wie beispielsweise Ablaufsimulationen oder spezifische Tabellenkalkulationsverfahren. Als Analysebasis dienen die Daten der Supply-Chain, die Unternehmensdaten mit Informationen über Zulieferer und Versorgungswege enthält. Eine Darstellungsform für Supply-Chains, die sich in der vergangenen Zeit bewehrt hat, ist das Data-Warehouse. Das Data-Warehouse stellt eine zentrale Datenbank zur Verfügung, in welcher sich alle entscheidenden Unternehmensdaten verbinden und zu Analysezwecken auswerten lassen [IDC17, S.1; HoNo09, S.1; Hin02, S.3].

Zuliefererdaten liegen meist in unterschiedlicher Dateiformaten und Quellen vor. Zudem können sie unterschiedliche Aggregationstiefe haben und Datenfehler aufweisen. Zur effizienten Nutzung müssen die Daten vor der Integration in die Supply-Chain von Fehlern bereinigt und auf die gewünschte Aggregationstiefe angereichert werden. Im allgemeinen Data-Warehouse-Kontext existieren unzählige Verfahren zur Anreicherung von Rohdaten oder zur Steigerung von Datenqualität bereits bestehender Data-Warehouse-Datensätze. Bisherige Literatur sieht jedoch kein genaues Vorgehen zur Fehlerbehebung und Anreicherung der, zur Integration in Supply-Chain-Systeme bestimmter, Daten vor.

Ziel dieser Arbeit ist ein Entwurf eines Ablaufmodells zur Überprüfung und Bereinigung von Zulieferer-Rohdaten vor der Integration in das Zielsystem. Die Rohdaten müssen vor der Transformation in ein Supply-Chain-Modell zunächst von Fehlern bereinigt und auf gewünschte Aggregationstiefe beschnitten oder angereichert werden. Voraussetzung dafür ist die Untersuchung bisheriger Verfahren zur Korrektur von Rohdaten und der nachträglichen Steigerung von Datenqualität in Data-Warehouse-Systemen. Darüber hinaus ist eine Aufarbeitung der Kategorisierungsmodelle von Supply-Chain-Daten für das Erreichen des Arbeitsziels nötig. [Hin02, S.3ff.]

Die Grundlagen für die spätere Untersuchung bildet die Aufarbeitung von Datenkategorien und die mögliche Verwendung im Kontext der Supply-Chain-Daten (Kapitel 2.1). Damit zusammengetragene Informationen führen anschließend zur Struktur des Data-Warehouse (Kapitel 2.2), in dessen Bezugsrahmen die Integration von Rohdaten erläutert und auftretende Datenfehler diskutiert werden.

Im Anschluss (Kapitel 3.1) werden grundlegende, Datenqualität mindernde Faktoren besprochen und unterschiedliche Datenfehler erläutert und kategorisiert. Um hinreichendes Verständnis über Schritte zur Fehlerüberprüfung und Korrektur von Dateninkonsistenzen zu erlangen, werden im Verlauf des Kapitels (Kapitel 3.2) bereits Anwendung findende Modelle zur Datenvorverarbeitung und Datenfehlerkorrektur besprochen.

Die Kenntnisse, die aus den bereits bestehenden Verfahren gezogen werden können, finden direkte Anwendung bei der Erstellung eines Konzepts zur Bereinigung von Rohdaten zur Verwendung in Supply-Chain-Systemen.

Die Anwendbarkeit des Datenkorrekturmodells wird mithilfe von Beispieldatensätzen validiert und das Resultat kritisch betrachtet. Das abschließende Fazit gibt Aufschluss auf die Umsetzbarkeit in Realanwendungen und auf mögliche Grenzen der Fehlerkorrektur.

2. Struktur und Darstellung von Supply Chain-Daten

In diesem Kapitel soll der Leser grundlegendes Verständnis über den Inhalt und die Darstellungsweise von Zuliefererdaten erhalten. Dazu wird zunächst untersucht, welche Daten in Zuliefererdatensätzen enthalten sind und wie sich Kategorisierungsmodelle auf diese anwenden lassen. Anschließend werden Darstellungsmöglichkeiten von Supply Chains beschrieben und erklärt, wie sich herkömmliche Supply-Chain-Daten integrieren lassen.

2.1 Dateneigenschaften und Kategorisierung

Das Ziel dieses Kapitels ist es, einen Einblick über bisher in der Literatur verwendete Kategorisierungsmodelle für Daten zu geben. Dabei wird der Fokus auf die Kategorisierung von Daten, die in Supply Chains vorkommen, gelegt. Die Kategorisierung von Supply-Chain-Daten ist im Supply Chain Management unumgänglich geworden, um die Daten effizient einzupflegen und analysieren zu können. Um dieses Ziel erreichen zu können, müssen zunächst Dateneigenschaften und Datentypen, der Daten, die in Supply Chains Anwendung finden, aufgearbeitet werden.

2.1.1 Dateneigenschaften

Eigenschaften geben an, wie Objekte beschaffen sind. Durch ihre Eigenschaften können Objekte in Relation zu anderen Objekten stehen oder eine Wechselwirkung haben. Erst über seine Eigenschaften kann ein Objekt einer Klasse oder einer Kategorie von zugehörigen Objekten zugewiesen werden. Sich nicht verändernde Eigenschaften eines Objektes werden letztlich als Merkmale bezeichnet. [Bol01]

Daten weisen ebenso Eigenschaften auf, über welche sie definiert werden. Die Literatur gibt eine breite Übersicht über Dateneigenschaften und den Kontext, für den sie Anwendung finden. Für den Kontext der Supply Chain Daten sind einige Dateneigenschaften weniger ausschlaggebend als andere. Aus diesem Grund werden im Verlauf dieses Abschnittes einige irrelevante Dateneigenschaften aussortiert.

Im Allgemeinen können Daten in zwei Repräsentationsformen auftreten. Entweder die Daten liegen in analoger oder in digitaler Form vor. In digitaler Form werden Folgen von Bits verwendet. Die Informationen werden mithilfe zweier Zustände, mit 0 und 1 dargestellt. Analoge Daten hingegen sind eine kontinuierliche Darstellung physikalischer Größen und können somit beliebige Werte annehmen. Maschinell interpretierbare Daten liegen immer in digitaler Form vor. Aus diesem Grund wird im weiteren Verlauf dieser Arbeit mit „Daten“ immer die digitale Repräsentationsform gemeint. Bezüglich der Darstellungsform wird zwischen zeichen- und bitorientierten Daten unterschieden. Bitorientierte Daten liegen direkt in ihrer endgültigen Form als nicht codierte Informationen vor. Durch sie werden z.B. Daten wie Bilder oder Audiodaten dargestellt. Dabei unterscheidet man nochmal zwischen statischen Daten wie Bildern und dynamischen Daten wie Videos oder Audiodaten. In Supply-Chain-Daten werden ausschließlich zeichenorientierte Daten verwendet. Sie liegen ebenfalls in 0 und 1 vor. Jedoch wird der Informationsgehalt erst durch die Kodierung klar. [Las06, S.216]

Zum Beispiel durch die Anwendung einer Zeichenkodierung wie ASCII werden die Bits der Bitfolge in Buchstaben, Zeichen und Ziffern übersetzt. Diese können formatiert, d.h. mit fester Länge oder strukturellen Vorgaben vorkommen. Ein Beispiel dafür sind Adressen. Diese bestehen immer aus Land, Stadt, Postleitzahl, Straße und Hausnummer. Jedoch gibt es auch unformatierte zeichenorientierte Daten. Diese werden für Daten verwendet, bei denen keine feste Struktur benötigt wird. Das können zum Beispiel zusätzliche Informationen als Text sein oder Datenfelder wie Emailadressen. [Las06, S.215f.] Eine Veranschaulichung der Darstellungsform als Baumstruktur zeigt Abb. 2.1.

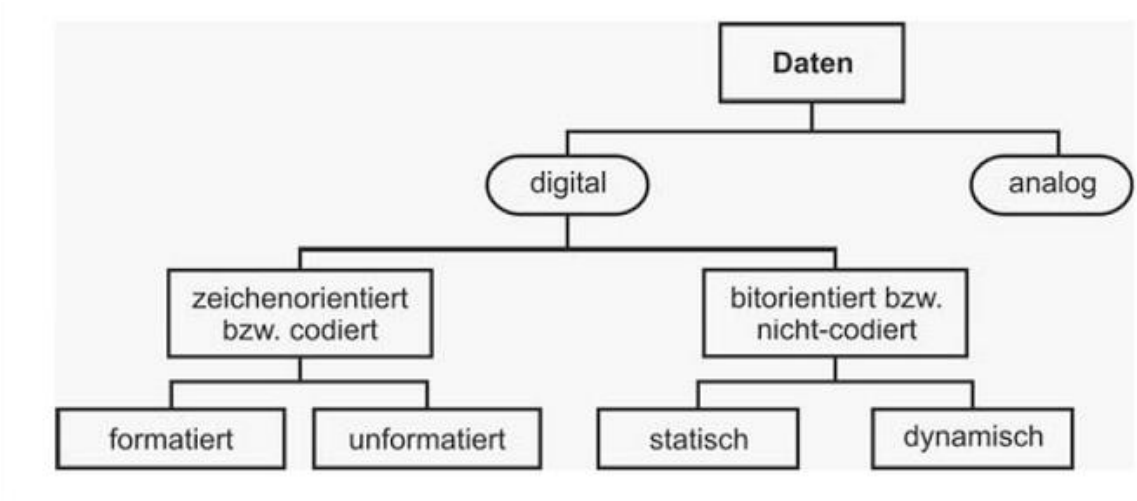


Abbildung 2.1: Darstellungsform von Daten, nach [Las06, S.216]

Zeichenorientierte Daten können letztlich noch nach der auftretenden Art der Zeichen differenziert werden. Es existieren numerische Daten (bestehend aus Zeichen und Ziffern), alphabetische Daten (bestehend aus Buchstaben und Sonderzeichen), alphanumerische Daten (bestehend aus Ziffern, Buchstaben und Sonderzeichen) und ikonische Daten (bestehend aus Bildzeichen). [Las06, S.216] Ein weiteres Klassifikationsmerkmal von Daten ist der Zweck. Daten können Primär- oder Sekundärdaten sein. Primäre Daten werden für bestimmte Aufgaben verwendet, wogegen sekundäre Daten zusätzlichen Aufgaben zugeteilt werden. [Las06, 218] Die Herkunft von Daten stellt eine weitere Eigenschaft dar. Daten können sowohl interne Daten sein, also ihren Ursprung im eigenen System haben, als auch externe Daten sein, die ihren Ursprung außerhalb des eigenen Systems haben. Ein Beispiel für interne Daten sind unternehmenseigene Daten, wie z.B. Bestandsdaten. Externe Daten hingegen sind zum Beispiel Zulieferbestände oder Frachtinformationen. In der Literatur wird häufig zwischen Inhaltsdaten und Metadaten unterschieden. Inhaltsdaten sind die informationshaltenden Daten. Metadaten im Gegensatz dienen zur Steuerung der Inhaltsdaten und geben die Struktur vor. Also bei einem Eintrag der Adressdaten wäre „Straße“ Teil der Metadaten und „Bahnhofstraße“ als Ausprägung Teil der Inhaltsdaten. [Pir11, S.146f.; Las06, S.219] Zudem haben Daten einen Zeitdauerbezug. Demnach existieren zustandsbezogene- und ereignisbezogene Daten. Zustandsbezogene Daten beinhalten Bestandsdaten und Stammdaten wie Lieferantenadressen. Ereignisbezogene Daten beziehen sich auf Ereignisse wie Transaktionen, aber auch Änderungen der Stammdaten, wie Adressänderungen. Des Weiteren unterscheidet man noch die Eigenschaften Funktionsbezug, in dem die Daten im Informationsverarbeitungsprozess zwischen Steuerungsdaten, Passivdaten, Ordnungsdaten und Identifikationsdaten unterschieden werden und der Stabilitätseigenschaft, die angibt, wie häufig sich die Daten im Zeitverlauf verändern. [Las06, S.218; Pir11, S.146f]

Anhand der Dateneigenschaften können die Daten kategorisiert werden. In seiner Ausarbeitung hat Johannes Ziegler erörtert, dass die einzigen, zur Kategorisierung dienlichen, Dateneigenschaften die Zeichenorientierung, die Zeichenart, der Funktionsbezug, die Stabilität und der Zeitdauerbezug sind. [Zie15, S.7] Im folgenden Abschnitt wird das von Ziegler(2015) erarbeitete Modell zur Kategorisierung von Supply-Chain-Daten vorgestellt und erklärt.

2.1.2 Kategorisierung von Supply-Chain-Daten

Die Kategorisierung von Daten spielt im Datenmanagement eine zentrale Rolle zur Erhaltung der Datenkonsistenz und einer hohen Datenqualität. Unternehmensdaten treten in verschiedenen Abteilungen meist in unterschiedlicher Form und Struktur auf, da Informationen unterschiedlich verarbeitet werden oder Daten unzureichend standardisiert sind. Ein Lösungsansatz zur unternehmensweiten Datenverwaltung wird durch die Bildung von Datenkategorien dargestellt. Dabei werden einzeln auftretende Daten anhand ihrer Eigenschaften und eines Kontextes in verschiedene Kategorien einsortiert. Da in der Supply-Chain unternehmensübergreifende Daten zusammenlaufen und zu Analyse- und Planungszwecken ausgewertet werden, ist eine unternehmensübergreifende Abstimmung der Datenkategorien notwendig.

Im Datenmanagement existieren verschieden Modelle und Ansätze zur Bildung von Datenkategorien. Jedes Modell versucht logische Unterscheidungen der Daten anhand ihrer Eigenschaften zu bilden und so Daten sinnvoll zu gruppieren. In Zieglers Arbeit wurden vier unterschiedliche Modelle überprüft und miteinander verglichen:

- Das Modell von Hansen/Neumann, das ihrem Grundlagenwerk zur Wirtschaftsinformatik (2005) entnommen ist
- Das Modell von Microsoft, das der Webseite des Unternehmens entstammt
- Das Modell von Chrisholm, das in Dirk Oedekovens Buch „Nutzenpotentiale hamonisierter Stammdaten“ (2011) Erwähnung findet
- Das Modell von Liebhart, das er 2010 in der Zeitschrift „Netzwoche“ erläuterte

Diese vier Kategorisierungsmodelle wiesen dabei alle einen ähnlichen Aufbau, wenn auch unterschiedliche Granularitätsgrade auf. Beim Vergleich kommt Ziegler zu dem Schluss, dass sich alle Modelle der Dateneigenschaften bedienen, die er zunächst als relevante Eigenschaften gekennzeichnet hatte [vgl. Abschn. 2.1.1]. Alle vier Modelle gliedern sich zunächst nach Zeichenorientierung. Im Anschluss daran wird nach Funktionsbezug in Nutz- und Steuerdaten unterschieden. Die Nutzdaten werden zum Abschluss nochmals in Zeitdauerbezug und Stabilität gegliedert. [Zie15, S.8ff]

Um nun allgemeingültige Datenkategorien auch auf Supply-Chain-Daten exakt anwenden zu können, muss zunächst geklärt werden, welche Art von Daten in Supply Chain vorkommen. Da der Fokus der Nutzung der Supply-Chain in der Planung und Optimierung der Wertschöpfungskette liegt, gehören in erster Linie Planungsprozesse wie die Beschaffungsplanung, Produktionsplanung, Absatzplanung und letztlich Transportplanung zu den Aufgaben der Supply Chain. Ziegler(2015) kommt zu folgenden Daten der Supply Chain, die in Abb.2.2 zu sehen sind und sich in Stamm- und Bewegungsdaten einteilen. [Zie15, S.19]

Stammdaten	Produktdaten
	Ressourcendaten
	Stücklisten
	Arbeitspläne
	Kundendaten
	Lieferantendaten
Bewegungsdaten	Lagerbestandsdaten
	Produktionsauftragsdaten
	Betriebsdaten

Abbildung 2.2: Stamm- und Bewegungsdaten der Supply-Chain [Zie15, S.19]

Mithilfe dieser vorkommenden Datentypen können nun im Anschluss in einer erweiterten Form der ursprünglichen Kategorisierungsmodelle Datenkategorien entwickelt werden, in die sich diese Daten exakt einordnen lassen. Dabei entstehen neun eindeutige Datenkategorien, die sich zunächst anhand der bekannten Dateneigenschaften gliedern (siehe Abb.2.3). Da Supply Chain Daten hauptsächlich Zuliefererinformationen, Angaben zu Waren und ihren Beständen oder Frachtinformationen in textueller oder numerischer Form enthalten, werden nur zeichenorientierte Daten berücksichtigt (vgl. Abschn. 2.1.1). Zunächst werden anhand der Zeichenorientierung formatierte und unformatierte Daten unterschieden. Unformatierte Daten werden direkt in die Kategorie „unstrukturierte Daten“ einsortiert, wobei es sich um Daten ohne feste Struktur oder vorgegebenes Format handelt (z.B. Emailadressen). Die formatierten Daten werden weiter unterteilt. Diese werden im Anschluss nach Funktionsbezug, also ihrem „Verwendungszweck“, in Nutz- und Steuerdaten unterschieden. Nutzdaten sind dabei die inhaltstragenden Daten, in denen die Informationen gespeichert sind und direkte Sachverhalte wiedergeben. Steuerdaten haben dabei keinen direkten informativen Nutzen. Sie tragen zur Steuerung der Informationsverarbeitungsprozesse bei und werden nochmal in zwei Kategorien gegliedert. Dient die Steuerdatei dazu Nutzdaten feste Strukturen zuzuweisen, handelt es sich um „Metadaten“. Wenn sie Verweise und Beziehungen zwischen Nutzdaten herstellen, werden sie der Kategorie „hierarchische Daten“ zugewiesen.

Nutzdaten werden weiter anhand des Zeitdauerbezugs gegliedert. Abwicklungsorientierte Nutzdaten werden dem Überbegriff „Transaktionsdaten“ zugewiesen. Sind sie hingegen zustandsorientiert, werden sie im Verlauf weiter unterteilt, um kategorisiert zu werden. Transaktionsdaten können ebenso wie die Steuerdaten anhand eines statischen Kontextes endgültig in Kategorien eingegliedert werden. Die Kategorie „Transaktionsaktivitätsdaten“ beinhaltet alle Daten, die durch Geschäftsaktivitäten anfallen, wie z.B. Bestellungen und Lieferungen. „Transaktionskontrolldaten“ sind hingegen Daten, die eben diese Aktivitäten protokollieren und nachvollziehbar machen.

Die Eigenschaft der Stabilität unterscheidet die zustandsorientierten Daten weiter. Sind die Daten variabel, werden sie direkt der Kategorie „Bestandsdaten“ zugeteilt, zu denen Lagerbestände oder Kapazitäten gezählt werden. Letztlich bilden die fixen zustandsorientierten Daten die letzten drei Kategorien, die im statischen Kontext der Stammdaten stehen: [Zie15, S.28ff]

- Die „Referenzdaten“, die standardisierte Abkürzungen für Länder, Bahnhöfe und Flughäfen enthalten.
- Die „Unternehmensstrukturdaten“, die den Aufbau des Unternehmens abbilden und ebenfalls eine Form von Stammdaten darstellen. Dazu gehören z.B. Kostenstellen.
- Die „Transaktionsstrukturdaten“, in die alle Daten über Produkte, Kunden und Lieferanten fallen.

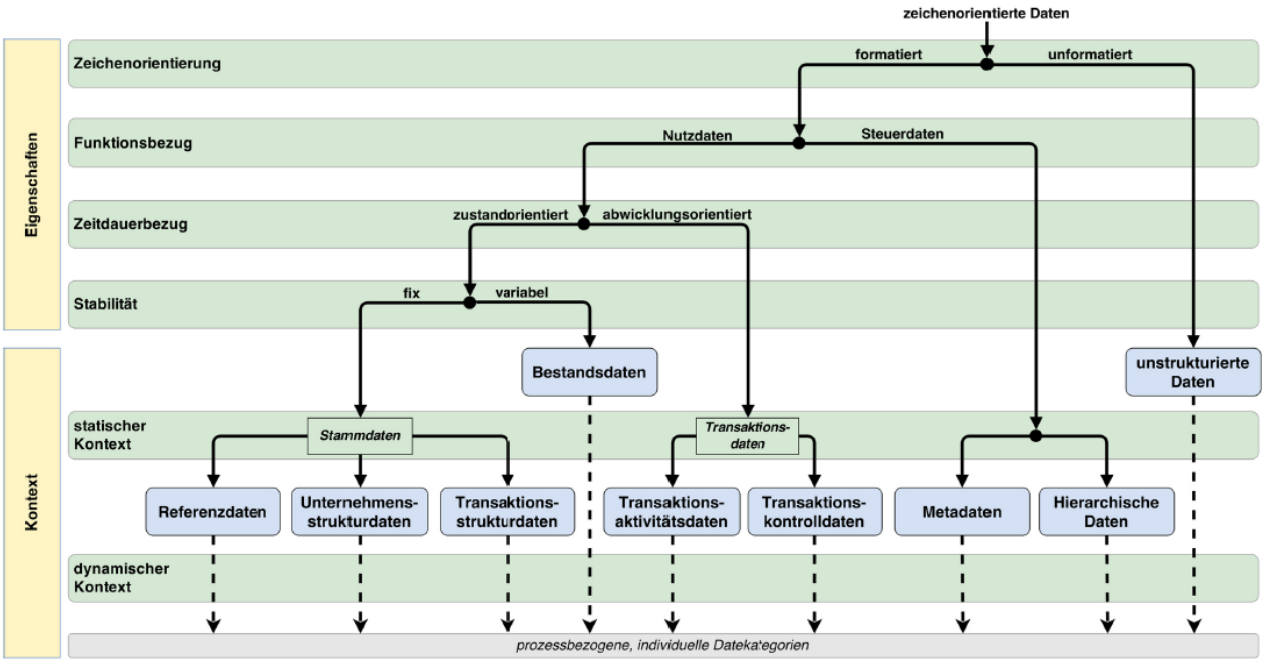


Abbildung 2.3: Kategorisierungsmodell für Supply-Chain-Daten nach Johannes Ziegler [Zie15, S.31]

Mithilfe des von Ziegler(2015) beschriebenen Modells lassen sich Daten, die in Supply Chains anfallen, eindeutig einer Datenkategorie zuweisen. Die Möglichkeit, diese Datenkategorien zu bilden, trägt erheblich dazu bei, fehlerhafte Supply-Chain-Daten zu erkennen und nach Möglichkeit zu bereinigen.

2.1.3 Aufbau und Funktion einer Supply-Chain

Die Analyse der Daten der Supply Chain ist für produzierende Unternehmen ein Hilfsmittel zur Organisation und Abstimmung aller Prozesse in der Wertschöpfungskette mit dem Ziel, diese zügig, effizient und effektiv zu gestalten. Bei der Supply Chain handelt es sich um ein komplexes Netzwerk von Beziehungen zwischen Planung, Steuerung, Beschaffung, Produktion, Lieferung und Kontrolle. Die Daten der Supply-Chain sollen dabei helfen, Zulieferer zu organisieren, Lieferzeiten gering zu halten und Lagerbestände zu reduzieren. Die Supply-Chain-Daten werden dabei nicht nur zur Planung von anstehenden Transaktionen verwendet, sondern auch zur Analyse vergangener Prozesse herangezogen. Dafür müssen die Daten korrekt, aktuell, zeitnah verbucht, konsistent, vollständig und redundanzfrei sein. [Zie15, S.15; Hil04, S.239]

Die von der Supply Chain verarbeiteten Daten werden durch verschiedenste unternehmensinterne und unternehmensübergreifende IT-Systeme verarbeitet [Zie15, S.16]. Die heutigen IT-Systeme, die in der Supply Chain Anwendung finden, sind ein Ergebnis jahrzehntelanger Erweiterung und Weiterentwicklung. Aus historischer Sicht haben die heutigen IT-Systeme ihren Ursprung in einem Programm zur Produktionssteuerung und -planung. In den 1960er Jahren wurde eine Lösung entwickelt, die die Materialbedarfsplanung unterstützen sollte. Das Resultat, das *Material Requirement Planning* (kurz: *MRP*) hat erstmals den Bedarf an Einzelteilen, Rohstoffen und Unterbaugruppen anhand der zu fertigenden Endprodukte und nicht nach der Lagerhaltung gerichtet. Das hatte eine deutliche Effizienzsteigerung der Materialbedarfsplanung zur Folge. [Zie15, S.16; Vah1; Wik1] Eine Weiterentwicklung dieses Systems, um die Planung über sämtliche Teilbereiche der Produktionsplanung zu ermöglichen, folgte etwa 1970. Ab diesem Zeitpunkt wurden zusätzlich Produktionskapazitäten mit in die Planung einbezogen. Zu diesem Zweck wurde *MRP* zur *Produktions-Planung und Steuerung (PPS)* erweitert. Diese wird auch als *MRPII* bezeichnet. Das Ziel dabei war, die Produktion möglichst stark auszulasten, um eine hohe Effizienz in der Fertigung zu erreichen. [Zie15, S.16; Vah1]

Durch die Integration der übrigen Unternehmensbereiche in die Planung und Steuerung in das MRPII-System sind letztlich die *ERP*-Systeme entstanden (*Enterprise Resource Planning*). Im Gegensatz zu genannten Vorgängern, konnten ab diesem Punkt die Daten nicht nur im industriellen Bereich genutzt werden, sondern waren auch in Unternehmensabteilungen, wie Finanzen oder Controlling nutzbar. Durch die Vernetzung von Unternehmen in Form von Supply Chains, ist die reine unternehmensbezogenen Planung und Steuerung, wie durch ERP-Systeme, nicht mehr ausreichend. Eine Planung nach dem „Top-Down“ Prinzip ist für die unternehmensübergreifende Planung ungeeignet, da bei diesem Prinzip der Planungsprozess erst im Verlauf der Planungsebenen an Details gewinnt. D.h., dass zu Beginn des Planungsprozesses spätere Komplikationen wie Liefer- oder Produktionsengpässe nicht bekannt sind. Eine Lösung für dieses Problem schuf die Einführung von *APS*-Systemen (*Advanced Planing and Scheduling*). *APS*-Systeme integrieren klassische *ERP*-Systeme und ergänzen ihre Ergebnisse durch fortgeschrittene Planungsverfahren in den unternehmensübergreifenden Geschäftsprozessen. [Zie15, S.17; Vah1] Abbildung 2.4 zeigt dazu grafisch den Zusammenhang zwischen den in der Supply Chain Verwendung findenden IT-Systemen.

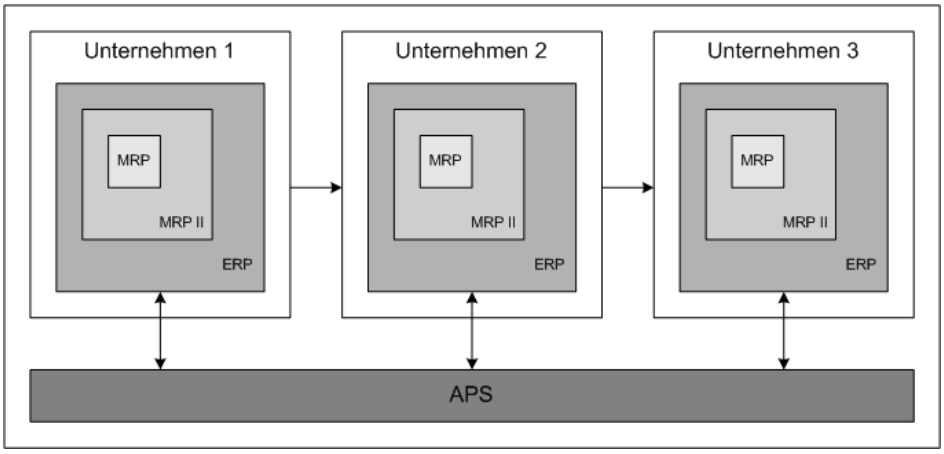


Abbildung 2.4 Zusammenhang der IT-Systeme [Vah1]

2.2 Darstellungsform Data-Warehouse

Einem Unternehmen stehen vielfältige Möglichkeiten offen, wie eine Supply-Chain zu analytischen Zwecken und Planungszwecken ausgewertet werden kann. Eine Möglichkeit, die sich in der Vergangenheit durchgesetzt und bewährt hat, ist das Data-Warehouse. In diesem Kapitel soll der Leser einen Einblick in die Verwendung eines Data-Warehouse bekommen. Hierzu werden zunächst die grundlegenden Darstellungsformen vorgestellt und in den direkten Kontext der Supply Chain gesetzt. Anschließend werden einige Organisationsformen besprochen und gegenübergestellt. Abschließend soll ein Einblick in das Gebiet der Datenversorgung und Informationsintegration in solchen Systemen geschaffen werden.

2.2.1 Konzept und Begriffsdefinition

Als häufigste Definition für das Data-Warehouse ist die von William Harvey Inmon (2005), der als Begründer des Data-Warehouse-Konzeptes gilt, zu finden:

„A data warehouse is a subject-oriented, integrated, nonvolatile, and time-variant a collection of data in support of management's decisions." [Inm05, S.29]

Definitionsgemäß stellt das Data-Warehouse eine Sammlung der Unternehmensdaten zur Unterstützung von Managemententscheidungen bereit. Nach Inmon sind diese Daten themenorientiert, integriert, beständig und zeitbezogen. Bis heute wurde diese Definition häufig von anderen Autoren überarbeitet oder ergänzt. Allgemein lässt sich das Data-Warehouse als System zur redundanten Bereitstellung von Unternehmensdaten zu Analysezwecken zusammenfassen. Wichtig ist dabei, dass die Daten im Data-Warehouse aus heterogenen operativen Systemen zusammengetragen werden, jedoch mit reinem Lesezugriff in einer separaten Datenbank gehalten werden. Abbildung 2.5 veranschaulicht hierzu den konzeptionellen Aufbau eines Data-Warehouse

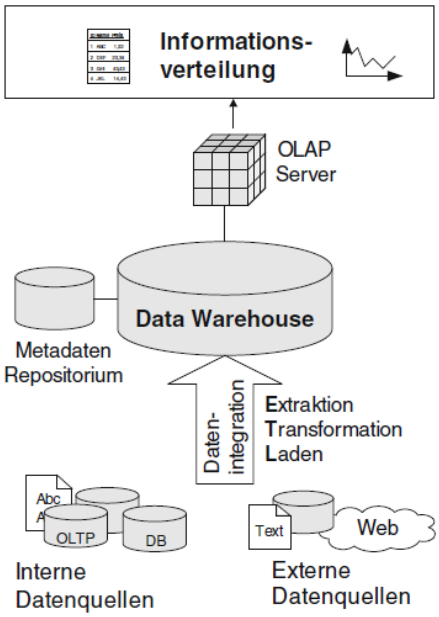


Abbildung 2.5: Data-Warehouse Konzept [Mül13, S.15]

Das Data-Warehouse zählt zu den Vertretern der materiellen Datenintegration. Das heißt, die Daten werden aus ihren ursprünglichen Quellen extrahiert und in neuer geeigneter Form abgespeichert. Als Quellen werden sowohl Unternehmensdaten (interne Daten), als auch Fremdquellen (externe Daten) verwendet (vgl. Abschn. 2.1.1). Bei der Umsetzung von Supply Chains kommen besonders die externen bzw. Fremddaten zum Tragen. Vorteil bei der Nutzung eines Data-Warehouse ist die redundante Zusammenführung und Aufbereitung heterogener Datenquellen, um eine nachfolgende Analyse zu vereinfachen oder gar erst zu ermöglichen. [Wro05, S.21]

Neben Realdaten, also den eigentlichen informationstragenden Daten, spielen die sogenannten Metadaten, die im Repositorium des Data-Warehouse hinterlegt sind, für den Aufbau und die Nutzung eines Data-Warehouse eine entscheidende Rolle, denn die Metadaten enthalten alle Informationen über die Realdaten. Sie beschreiben das Data-Warehouse-Schema und alle Restriktionen. [Mül13, S.13]

2.2.2 Datenversorgung

Die Datenbereitstellung und die Transformation neuer Daten in analysierbarer Form zur Nutzung im Data-Warehouse hat eine wesentliche Bedeutung für die spätere Analysierbarkeit und Aussagekraft der Informationen [Wro05, S.29]. Nach Inmon (2005) stellt die Datenintegration aus unterschiedlichen Quellen den wichtigsten aller Aspekte des Data-Warehouse dar [Inm05, S.30]. Er ergänzt, dass das Design und die Realisierung einer Datenspeisungsschnittstelle zur Transformation der Daten aus operativen Systemen bis zu 80 Prozent des Arbeitsaufwandes bei der Erstellung eines neuen Data-Warehouse ausmachen kann.

Dies liegt vor allem an der vielfältigen Heterogenität der Datenquellen [Inm05, S276]. Die Einspeisung von Daten aus den operativen Systemen wird oft mit dem Begriff ETL (Extract, Transform & Load) beschrieben.

Der ETL-Prozess besteht aus 3 unterschiedlichen Phasen. Die erste Phase beschreibt die Extraktion der Rohdaten aus ihren ursprünglichen Datenquellen. Diese Daten werden zunächst in einem temporären Arbeitsbereich geladen. Die Daten verbleiben nur solange im Arbeitsbereich, bis sie vollständig in die Datenbank geladen worden sind. In der zweiten Phase, der Transformation, werden die neuen, noch heterogene Daten in eine für das Data-Warehouse passende syntaktische Form gebracht [Ven15, S.37].

Die Autoren Kemper und Finger (1998) fassen die Prozesse der Extraktion und Transformation unter dem Begriff der Transformation zusammen und unterteilen den Prozess durch die Sub-Prozesse Filterung, Harmonisierung, Verdichtung und Anreicherung.

Die Filterung erfüllt dabei den Arbeitsschritt der Extraktion. Zusätzlich werden die Daten von syntaktischen und inhaltlichen Defekten bereinigt. Einige dieser Fehler werden automatisch erkannt und bereinigt. Darunter fallen systematische Fehler, wie nicht interpretierbare Steuerzeichen oder uneinheitliche Zeichensätze. Andere Fehler, die zwar erkannt werden, aber nicht automatisch bereinigt werden können, werden in Logdateien vermerkt. [KeFi98, S.77]

Die Harmonisierung bildet den anschließenden Prozess. In diesem Schritt werden die neuen Daten von Heterogenität befreit. Dies beinhaltet die Beseitigung von unterschiedlichen Homonymen und Synonymen. Abschließend werde die Daten themenbezogenen Gruppen, wie Kundenstammdaten oder Produkten zugewiesen. [KeFi98, S.69] Dafür nötige Daten werden über einen Kontrolldatenfluss aus dem Metadatenmanagement bzw. dem Repositorium bezogen [Ven15, S.38].

Im dritten Schritt, der Verdichtung, liegen die Daten nun in niedrigster Granularitätsebene vor. Diese müssen anschließend mit einem geeigneten Verdichtungsprozess auf benötigte Aggregationstiefe gebracht werden. Eine Berechnung der benötigten Aggregation im System ist aus Performancegründen weniger sinnvoll. [KeFi98, S.72]

Während der Anreicherung, die den letzten Schritt der Transformation nach Kemper und Finger (1998) bildet, werden die harmonisierten und angereicherten Daten mit betriebswirtschaftlichen Kennzahlen wie Abweichungen oder Mittelwerten ergänzt. Dadurch wird ebenfalls die spätere Performance im Data-Warehouse gesteigert, weil diese Werte nicht erst bei jeder Abfrage erneut erstellt werden müssen.

Der abschließende Schritt des ETL- Prozesses stellt das Laden da. In diesem finalen Schritt werden die nun fertig angepassten Daten in die eigentliche Basisdatenbank des Data-Warehouse geladen.

2.2.3 Datenbereitstellung zur Integration bestimmter Unternehmensdaten

Die Grundlage der analytischen Systeme wie Supply Chains bilden die Unternehmensdaten. Unternehmensdaten liegen dafür meist in unterschiedlichen Datenformaten und Datenquellen vor. Dabei zeigen Studien, dass bei der Wahl der Datenhaltungswerkzeuge nach wie vor Microsoft Excel das meist genutzte Programm ist. Eine Unternehmensbefragung des Business Application Research Centers (BARC) hat 2016 noch ergeben, dass selbst zu Unternehmensplanungen oftmals nur Microsoft Excel als Planungswerkzeug Anwendung findet und operative Systeme oder gar Business-Intelligence-Werkzeuge, wie sie in Data-Warehouses verwendet werden, nur in den seltensten Fällen berücksichtigt werden.

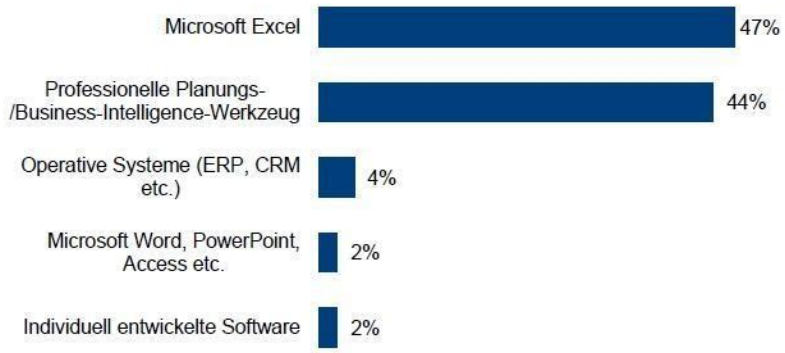


Abbildung 2.6: Zur Unternehmensplanung genutzte Systeme [COW]

Dazu kommt, dass laut BARC ein Viertel der Unternehmen nicht mal alle Daten im selben System mit identischer Struktur umsetzen. Vielen falls sind auch gleichzeitig unterschiedliche Werkzeuge parallel im Einsatz, wodurch die Komplexität bei der Zusammenführung der Daten unnötig erhöht wird. [COW]

3. Datenqualität und Fehlerfreiheit

Im 2. Kapitel wurden der grundlegende Aufbau und die Funktion von Supply-Chain-Datenbanken behandelt. Ebenso wurde die Kategorisierung und die Transformation von Unternehmensdaten und Fremddaten zur Nutzung in Supply-Chains thematisiert. Die Fehlerbehaftung und die Korrektur von Fehlern und von Heterogenität, die in Abschnitt 2.2.2 angesprochen wurde, wird im Laufe dieses Kapitels aufgegriffen und detailliert besprochen. Abschließend werden einige Korrekturverfahren zu zuvor erläuterten Fehlertypen vorgestellt.

3.1 Datenfehler und Heterogenität

In diesem Kapitel wird das Ziel verfolgt, dem Leser die Problematik von Datenheterogenität und Dateninkonsistenzen, wie Datenfehler nahe zu bringen. Datenfehler mindern immer die Datenqualität und haben somit direkte Auswirkungen auf die Nutzbarkeit der Daten. Dabei wird zunächst geklärt, was Heterogenität bedeutet und welche Formen Heterogenität zwischen Datenquellen annehmen kann bzw. durch welche Ursachen sie hervorgerufen wird. Um die genauen Konsequenzen von Fehlern in Datensätzen, die zu Analyse- oder Planungszwecken herangezogen werden, überblicken zu können, müssen abschließend die verschiedenen Arten von Fehlern besprochen werden.

3.1.1 Heterogenität

Heterogenität ist ein immer wieder auftauchendes Problem, wenn es um die Zusammenführung zuvor getrennter Datensätze geht. Mit der Bereitstellung von Datenressourcen aus unterschiedlichen Quellen (seien es Datenbanken oder andere Datenmodelle) treten immer wieder unterschiedliche Formen der Heterogenität auf. In der Literatur gibt es unterschiedlichste Klassifikationen von Heterogenität. Da Quellen für Datenhaltungsmodelle wie Data Warehouse oder im Hinblick auf das Thema dieser Arbeit für Supply-Chain vielzählig sein können, kann sich an diesem Punkt schon der erste Fall von Heterogenität befinden. Bei der Modellierung von Datenbanken stehen verschiedene Modelle zur Verfügung, wie relationalen Datenbanken, Netzwerkdatenmodelle oder objektorientierte Datenmodelle. Jedes dieser Modelle stellt dabei unterschiedliche Semantiken und Strukturen zur Verfügung und bildet unterschiedliche Modellierungskonzepte. So unterstützt z.B. das objektorientierte Datenmodell Generalisierung oder Vererbung, wogegen diese Konzepte im relationalen Datenbankenmodell nicht vorgesehen sind. Des Weiteren gibt es Unterschiede in den verwendeten Anfragesprachen der Datenmodelle. Aufgrund dessen können erhebliche strukturelle Unterschiede zwischen den Datenmodellen herrschen, auch wenn diese den gleichen Sachverhalt darstellen. [Wro05, S.16; Schna04, S.6; LeNa07, S.58ff.]

Auf der Ebene der Schemata kann ebenfalls bereits Heterogenität auftreten. Selbst wenn den Schemata einheitlichen Datenquellen zugrunde liegen, kann eine unterschiedliche Auffassung der Informationen oder eine unterschiedliche Informationsnutzung bei der Modellierung zu Heterogenität führen. Bei der Zusammenführung aus unterschiedlichen heterogenen Quellsystemen kommt es daher häufig zur sogenannter *semantischen Heterogenität*. Die Schemata der Quellsysteme bestehen dabei aus Relationen und Attributen, wobei jedes Attribut einen Namen und eine Semantik besitzt. Bei der Zusammenführung der Schemata enthalten diese nur die Namen.

Erst durch die Interpretation der Namen muss sich die Semantik rekonstruieren lassen. Dabei treten unterschiedliche Unklarheiten auf (siehe Abb. 3.1).

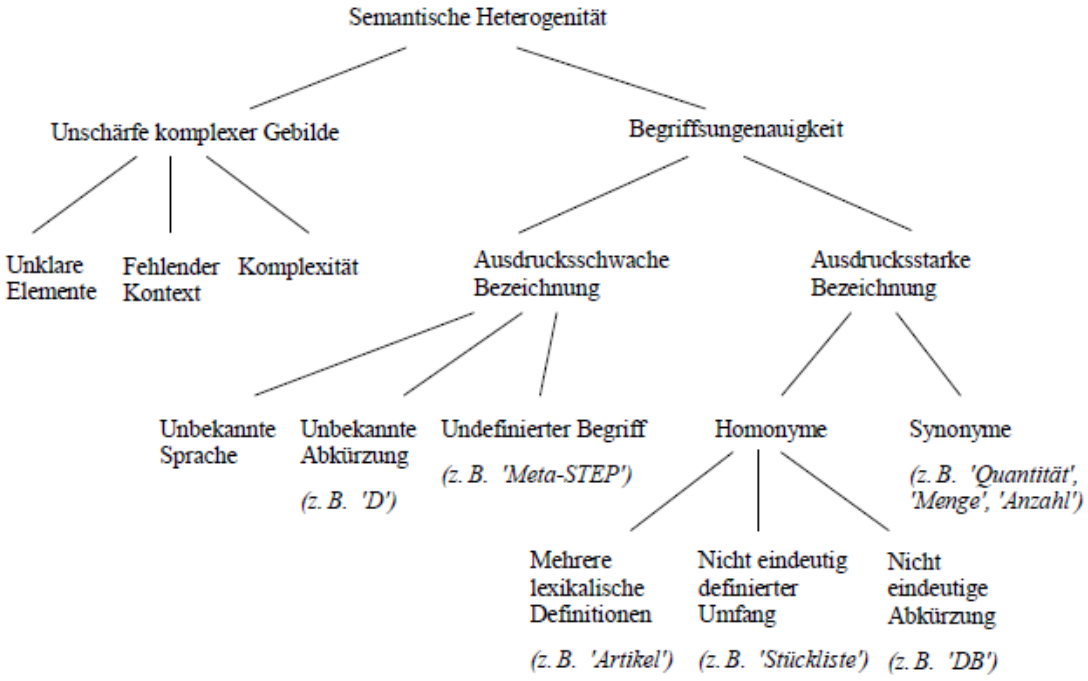


Abbildung 3.1: Baum-Darstellung Semantischer Heterogenität [Schna04, S.7]

Diese Unklarheiten können komplette Gebilde oder Strukturen betreffen (*Unschärfe komplexer Gebilde*) oder einzelne Begriffe (*Begriffsungenauigkeit*). Die Unschärfe komplexer Gebilde kann das gesamte Schema betreffen, sodass es durch mangelndes Verständnis der zentralen Begriffe oder durch fehlenden Zusammenhang zwischen Elementen nur teilweise oder gar nicht verstanden werden kann. *Begriffsungenauigkeit* kann in *ausdrucksschwache* und *ausdrucksstarke Bezeichnungen* unterschieden werden. *Ausdrucksstarke Bezeichnungen* sind entweder unterschiedliche Bezeichnungen mit gleicher Bedeutung (*Synonyme*) oder gleiche Bezeichnungen mit unterschiedlicher Bedeutung (*Homonyme*). In den Schemata der Abbildung 3.2 werden diese Sachverhalte illustriert. In Schema (a) wird unter dem Attribut „Kontakt“ die Emailadresse des Kunden gespeichert. In Schema (b) hingegen das Datum der letzten Kontaktaufnahme mit dem Kunden. Die Namen dieser Attribute sind homonym. Zudem sind in beiden Schemata die Straßen der Kundenadresse gespeichert. In Schema (a) unter dem Attribut „Strasse“, in Schema (b) hingegen unter der synonymen Attributbezeichnung „Kunden_Str“.

(a)

Name	Kontakt	Strasse
Müller	mueller@abc.com	Hohestr.
Schmidt	schmidt@xy.de	Breitestr.

(b)

Name	Kontakt	Kunden_Str
Huber	21.03.2003	Lenaustr.
Hinze	10.12.2003	Goethestr.

Abbildung 3.2: Darstellung von homonymen und synonymen Begriffsungenauigkeiten [Wro05, S.17]

Ausdrucksstarke Begriffe lassen sich in der Regel interpretieren, da diese ein gewisses Grundverständnis der Begrifflichkeiten zulassen. Im Gegensatz dazu bestehen *ausdrucksschwache Begriffe* aus unzureichend oder gar nicht definierten Begriffen. Ein Beispiel dafür sind *undefinierte Begriffe* wie „Meta-STEP“, *unklare Abkürzungen* wie „D“ oder Einträge in *unbekannter Sprache*. [Wro05, S.16ff; Schna04, S.6]

Ebenfalls kann der Einsatz unterschiedlicher Modellierungskonzepte des Datenmodells zu unterschiedlicher Struktur führen. Es entstehen unzählige Möglichkeiten, denselben Sachverhalt unterschiedlich darzustellen. Diese Heterogenität wird als strukturelle Heterogenität bezeichnet. Beispielsweise ist es möglich, einen Sachverhalt entweder in einer Relation oder in einem Attribut zu modellieren. Abbildung 3.3 zeigt die Modellierung von Personen und zugehörigem Geschlecht. In Schema A wurden zwei Tabellen zur Unterscheidung von männlichen und weiblichen Personen verwendet. In Schema B wurde nur eine Tabelle verwendet, in der die Unterscheidung innerhalb von Attributen stattfindet. [Schna04, S.8]

Tabelle Männer (Schema A)

Vorname	Nachname
Peter	Meier

Tabelle Frauen (Schema A)

Vorname	Nachname
Eva	Klein

Tabelle Personen (Schema B)

Vorname	Nachname	männlich	weiblich
Peter	Meier	X	
Eva	Klein		X

Abbildung 3.3: Strukturelle Heterogenität – Relation/Attribut [Schna04, S.8]

Derselbe Sachverhalt lässt sich wiederum auch über Attributwerte modellieren. Abbildung 3.4 zeigt die Unterscheidung zwischen der Modellierung als Attribut und der Modellierung als Wert eines Attributes. In Schema A sind wieder die Ausprägungen in Form der Attribute gekennzeichnet, in Schema B hingegen durch unterschiedliche Werte des Attributs „Geschlecht“. [Schna04, S.8]

Tabelle Personen (Schema A)

Vorname	Nachname	männlich	weiblich
Peter	Meier	X	
Eva	Klein		X

Tabelle Personen (Schema B)

Vorname	Nachname	Geschlecht
Peter	Meier	männlich
Eva	Klein	weiblich

Abbildung 3.4: Strukturelle Heterogenität – Attribut/Wert d. Attributs

Die Verwendung von unterschiedlichen Elementen zur Modellierung desselben Sachverhaltes in einem Modell wird schematische Heterogenität genannt. Schematische Heterogenität wird dabei als Spezialform der strukturellen Heterogenität aufgefasst. [Schna04, S.8]

Schließlich können auch Werte, die jeweils die identische Eigenschaft in gleicher Struktur beschreiben unterschiedliche Modellierung aufweisen. So kann die Speicherung einer Eigenschaft in unterschiedlichen Datenquellen in differenzierten Datentypen stattfinden. So können Straßennamen aus Adressdaten verschieden dargestellt werden: Z.B. „Bahnhofstraße“ durch alternative Darstellungen wie „Bahnhof Straße“, „Bahnhof Str.“, Bahnhof-Straße“ oder „Bahnhof-Str.“

Ferner können auch unterschiedliche Maßeinheiten zu Heterogenität führen. Die Nutzung von unterschiedlichen Temperaturangaben (Celsius oder Fahrenheit), unterschiedliche Nutzung von Gewichts- und Maßeinheiten verschiedener Länder oder Längenmaße in Zentimeter und Zoll können bei der Zusammenführung zu Komplikationen beitragen. Des Weiteren können noch unterschiedlich gewählte Genauigkeiten, also z.B. die Zahl der Nachkommastellen oder eine unterschiedliche Repräsentationsform beitragender Faktor für Heterogenität sein. [Wro05, S.17ff; LeNa07, S.321ff.]

Dies sind alles nur einige Bespiele für Heterogenität, die zwischen unterschiedlichen Datenquellen auftreten können. Letztlich gibt es unzählige Möglichkeiten, die es erschweren, Datenquellen zusammenzuführen, wenn diese zuvor separiert voneinander existierten.

3.1.2 Datenfehlertypen und Klassifikation

Nicht nur Heterogenität ist ein Kriterium, das Integration von Daten zu komplexen und zeitintensiven Vorhaben wachsen lässt. Datenquellen sind nur in seltenen Fällen fehlerfrei und müssen, bevor ihre Informationen in das zukünftige Datenmodell integriert werden können, von Fehlern befreit werden.

Einige Fehler entstehen unter anderem bei der Integration von neuen Datenquellen, andere Fehler sind auf mangelnde Aktualität zurückzuführen. Im Folgenden werden einige Fehlertypen und ihre Herkunft besprochen.

Bei auftretenden Fehlern unterscheidet man zwischen einfachen und schwerwiegenden Fehlern. Einfache Fehler können durch das Betrachten eines einzelnen Tupels erkannt werden. Schwerwiegende Fehler können nur durch die Betrachtung mehrerer Tupel in ihrem Zusammenhang erkannt werden.

Fehler lassen sich zusätzlich anhand ihres Vorkommens klassifizieren. Die erste Klasse der Fehler besteht bereits in den *einzelnen Datenquellen*. Die zweite Klasse hingegen wird erst bei der Integration von mehreren Datenbeständen sichtbar. Beide Klassen lassen sich in „Bezug auf das Schema fehlerhaft“ und in „in sich fehlerhaft“ unterteilen. Abbildung 3.5 zeigt eine Baumdarstellung der Fehlerklassen, auf dessen Ausprägungen im Folgenden eingegangen wird. [LeNa07, S.317, Mül13, S.41ff.; RaDo00, S.5]

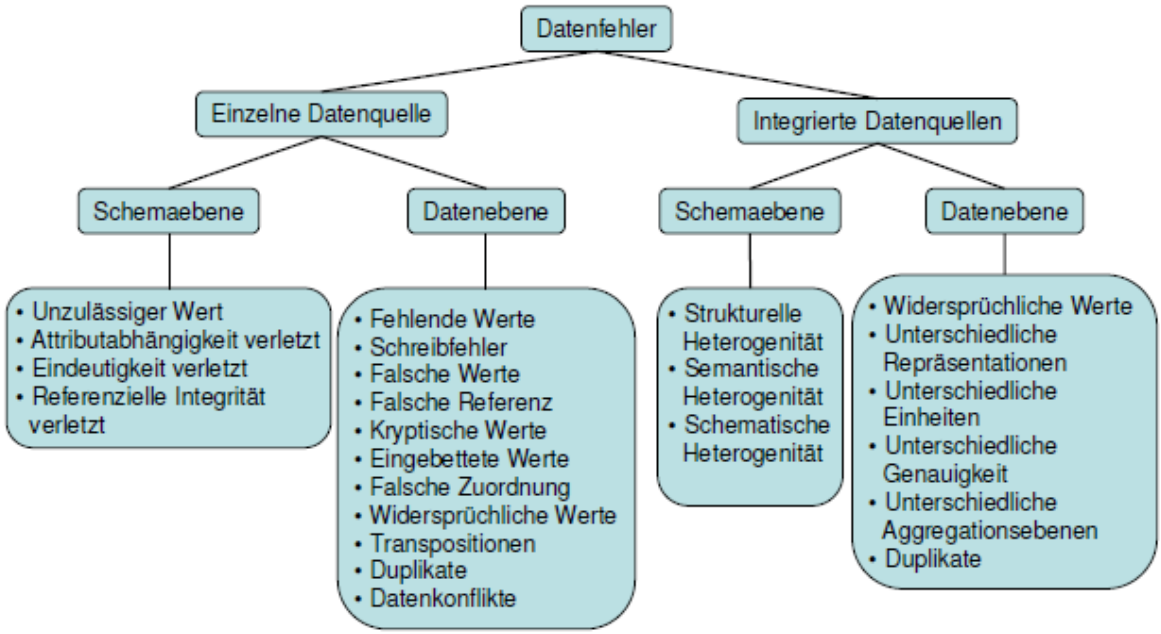


Abbildung 3.5: Klassifikation von Datenfehlern [LeNa07, S.319]

Fehler *einzelner Datenquellen* auf *Schemaebene* betreffen vor allem Verletzungen der Integritätsbedingungen des Schemas. Integritätsbedingungen bilden ein wichtiges Hilfsmittel zur frühzeitigen Erkennung von Datenfehlern. *Unzulässige Werte* stellen dabei die erste Fehlerklasse dar. Diese treten auf, wenn z.B. Datenwerte außerhalb der angegebenen Domäne liegen. Das können z.B. fehlerhafte Datumsangaben wie 31.02.17 sein oder negative Gewichtsangaben wie -250Kg. Fehler können auch Abhängigkeiten zwischen verschieden Attributen, die nicht eingehalten werden, sein. Wenn beispielsweise das angegebene Alter nicht mit dem Geburtsdatum übereinstimmt oder Kalenderdaten nicht mit dem passenden Wochentag, spricht man von *Verletzungen der Attributabhängigkeit*. Wenn Datenwerte als eindeutig gekennzeichnet sind, aber mehrfach vorkommen, spricht man von Datenfehlern, die die *Eindeutigkeit verletzen*. Ebenso werden Daten als fehlerhaft bezeichnet, wenn sie auf einen Fremdschlüssel verweisen, der in der Quelldatei nicht vorhanden ist. In diesem Fall wird die *referenzielle Integrität verletzt*. [LeNa07, S.319; Mül13, S.38; RaDo00, S.5ff]

Datenfehler einzelner Datenquellen auf Datenebene können eine deutlich größere Anzahl an Ausprägungen annehmen. Sie werden nicht durch Spezifikationen auf Schemaebene verhindert. *Fehlende Werte* stellen den ersten Fehlertyp dar. Diese Attributwerte werden vom System als „null“ gekennzeichnet. Grund für fehlende Attribute im System sind meist Nachlässigkeit beim Erstellen der Quelldatei oder fehlende Informationen. Viele Systeme unterbinden mittlerweile, dass Werteingaben frei bleiben können. Da viele Daten weiterhin ihren Ursprung in manueller Eingabe haben, werden solche Pflichteingaben durch „Dummywerte“ (z.B. „123“ oder „abc“) umgangen. Dies erschwert zusätzlich das Auffinden fehlender Werte, da es nicht mehr möglich ist, lediglich alle „Null-Werte“ zu untersuchen. [LeNa07, S.320; RaDo00, S.5]

Auch *Schreibfehler* können ihren Ursprung in der manuellen Eingabe von Daten haben, aber auch durch automatische Schrifterkennung oder das Parsen von Daten entstehen. Schreibfehler müssen in der Regel durch Domänenexperten händisch erkannt und behoben werden. Ein weiterer Fehler der sich äußerst schwer nachvollziehen lässt, sind *falsche Werte.*

Falsche Werte entsprechen nicht den reellen Gegebenheiten, können allerdings meist nur entdeckt und behoben werden, wenn der entsprechende Realwert bekannt ist. Wenn ein Fremdschlüssel auf eben solch einen falschen Wert verweist, spricht man von einer *falschen Referenz.* In vielen Fällen werden Abkürzungen benutzt, um Firmennamen, Bundesländer, Materialien, etc. darzustellen. Wenn Abkürzungen jedoch unbekannt sind oder sie keinen Rückschluss mehr auf den eigentlichen Wert geben, handelt es sich um einen *kryptischen Wert*. Da Quellsysteme oftmals händischen Ursprung haben, kann es dazu kommen, dass im verwendeten Schema des Quellsystems nicht für alle Informationen auch Eintragsmöglichkeiten vorgesehen sind. Dies betrifft oftmals zweite Vornamen bei Personen, Einfahrtsnummern bei Großlagern oder andere zusätzliche Informationen. Diese werden in vereinzelten Fällen in Attributen, die für andere Angaben vorgesehen sind, beigefügt. Z.B. (Vorname = „Andreas Michael“) oder (Hausnummer = „15 Einfahrt 4A). Dieser Fehlertyp wird *eingebetteter Wert* genannt. In ähnlichem Rahmen können auch Werte im falschen Attribut landen, wie etwa eine Vertauschung von Vor- und Nachname bei der Eingabe. Dies wird *falsche Zuordnung* genannt. Ein durchaus häufig vorkommender Fehler ist die *Widersprüchlichkeit von Werten.* Dieser Fehler entsteht, wenn Werte im Schema nicht zueinander passen (z.B. in Adressdaten, wenn eine Straße im Postleitzahlgebiet nicht existiert). Sind die Abhängigkeiten der Werte im Schema nicht definiert, muss zur Entdeckung und Bereinigung dieser Fehler Mehraufwand betrieben werden. Ein Fehler, der in Datenbanken häufig vorkommt, ist das *Duplikat.* Duplikate sind im Grunde Kopien desselben Datensatzes aus unterschiedlichen Datenquellen. Sollten zwei oder mehrere Datenquellen integriert werden, in denen derselbe Datensatz einer Person oder einer Firma gespeichert war, kommt es in der integrierten Datenbank zu Duplikaten. Sollten die Datensätze geringfügige Abweichungen voneinander haben, wie z.B. ein einziges Attribut, das diese unterscheidet, nennt man diesen Widerspruch zwischen den Duplikaten *Datenkonflikt*. [LeNa07, S.320ff.; Mül13, S.38ff.; RaDo, S.6] Abbildung 3.6 zeigt einige Fehlerausprägungen anhand verschiedener Einträge in einem Schema.

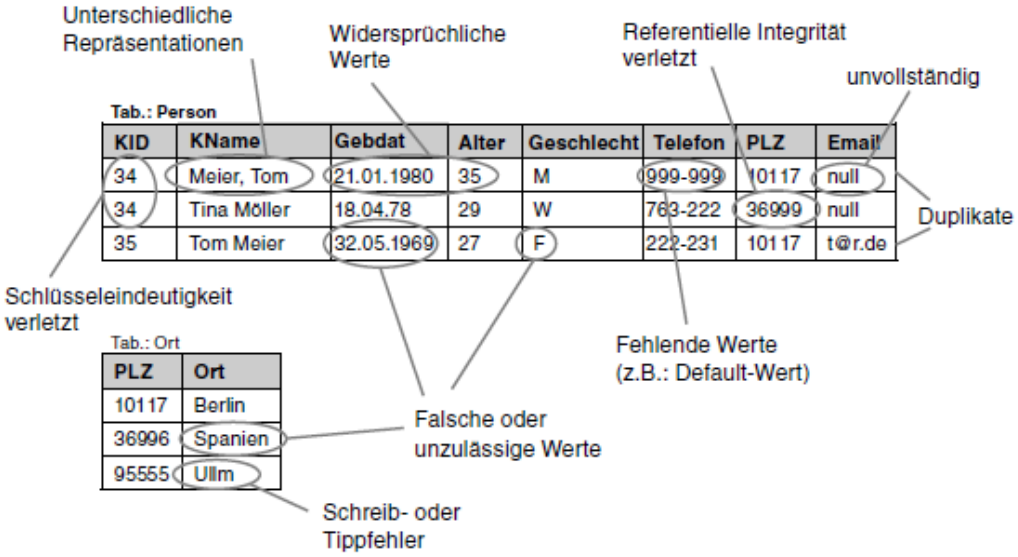


Abbildung 3.6: Beispielhafte Darstellung von Datenfehlern [Mül13, S40]

Neben Fehlern in den einzelnen Daten können auch Fehler in integrierten Datenquellen entstehen. Diese Fehler wurden bereits größtenteils in Abschnitt 3.1.2 besprochen. Auf Schemaebene handelt es sich um Heterogenität zugrundeliegender Fehler. Auf Datenebene können grundsätzlich dieselben Fehlertypen wie in den einzelnen Datenquellen auftreten.

Zusätzlich kann die Wahl von Einheitsmaßen, Genauigkeiten oder Repräsentationsformen für Datenfehler sorgen. (vgl. Abschn. 3.1.2) [LeNa07, S.321ff.]

3.2 Steigerung der Datenqualität

Nachdem im vorangegangenen Abschnitt auf die auftretenden Probleme bei der Datenintegration in eine zentrale Datenbank aufmerksam gemacht wurde, werden in diesem Abschnitt Konzepte zur Überwindung von Heterogenität und zur Beseitigung der zuvor erwähnten Datenfehlerklassen besprochen. Dabei wird zunächst auf die Heterogenität der einzelnen Datensätze eingegangen, um im Anschluss die Fehlerbehaftung der einzelnen Daten zu prüfen und auftretende Fehler zu beseitigen. Dies entspricht dem Vorgehen bei der Realisierung integrierter Informationssysteme. [LeNa07, S.317]

3.2.1 Überwinden von Heterogenität

Dass die Zusammenführung von Datenquellen Konflikte zwischen den heterogenen Quelldaten verursachen kann, wurde bereits in Abschnitt 3.1.1 besprochen. Damit nun Daten aus Quellen, die unterschiedlichen Schemata und Datenmodellen unterliegen, dennoch zusammengeführt genutzt werden können, müssen diese auf eine gemeinsame Darstellung gebracht werden. Um ein brauchbares Endresultat zu erzeugen, müssen Gemeinsamkeiten der zu integrierenden Datensätze gefunden werden, die im Anschluss als Integrationspunkte dienen. Dieser Vorgang wird Schema-Integration genannt. Um die Daten aus den Quellen in die Datenbank des Data-Warehouse zu migrieren, muss eine Abbildung zwischen den Quellschemata und dem gewünschten Zielschema gefunden werden. Automatisierte Verfahren, die das bewerkstelligen, werden Schema-Matching-Verfahren genannt. [Schna04, S.9]

Schema-Matching-Verfahren benutzen die Operation Match. Diese Operation bekommt als Eingabe zwei Schemata, also beim Vergleich von Quelle A und Quelle B das Schema von A und das Schema von B. Als Ausgabe wird eine Abbildung zurückgegeben, die angibt, wie die Schemata ineinander überführbar sind. Diese Abbildung wird im Fachterminus „Mapping“ genannt. Das Mapping besteht dabei wiederum aus Mapping-Elementen, die die einzelnen Elemente der Schemata einander zuordnen. Dies geschieht über Mapping-Ausdrücke, die angeben, in welcher Beziehung die unterschiedlichen Schemaelemente stehen. Hierbei wird zwischen gerichteten und ungerichteten Ausdrücken unterschieden. Ein gerichteter Mapping-Ausdruck bildet ein Element des ersten Schemas genau auf ein anderes Element des zweiten Schemas ab. Ungerichtete Mapping-Ausdrücke hingegen können Beziehungen zwischen mehreren Elementen der Schemata sein. Mapping-Ausdrücke können demnach Vergleichsoperatoren, Funktionen wie Konkatenationen oder Mengenbeziehungen wie Teilmenge oder Schnittmenge enthalten. [Schna04, S.9ff.]

Im Allgemeinen lassen sich Matching-Verfahren anhand ihrer Vorgehensweise kategorisieren (siehe Abb. 3.7). Man unterscheidet zwischen einzelnen und kombinierten Matching-Verfahren, die weiter untergliedert werden können. Einzelne Verfahren werden in *schemabasiert* und *instanzenbasiert* unterteilt. *Schemabasierte* Verfahren greifen dabei auf die dem Schema zugrundeliegenden Informationen wie Schemaelemente, Datentypen und Beziehungen zurück. Die Informationen der eigentlichen Daten bleiben dabei unbeachtet.

Beim *instanzenbasierten* Matching wird auf die Daten selbst zurückgegriffen. Das macht immer dann Sinn, wenn zu wenig Informationen aus dem Schema selber hervorgehen oder falsche Interpretationen aus dem Schema möglich sind. [Schna05, S.11]

Weiter wird *elementbasiert* und *strukturbasiert* unterschieden. *Elementbasierte* Matching-Verfahren betrachten die einzelnen Elemente vom Kontext isoliert, wogegen *strukturbasierte* Verfahren den struturellen Kontext ohne Elemente analysieren. In der hier tiefsten Unterscheidungsebene können nun Übereinstimmungen der Schemaelemente anhand ihres Namens und einer in Textform befindlichen Beschreibung gefunden werden. Dieses Vorgehen wird *sprachbasiert* genannt. Schemata enthalten häufig Integritätsbedingungen zur Definition von Datentypen, Bezeichnungen und Wertebereichen, die von *integritätsbasierten* Verfahren genutzt werden um Ähnlichkeiten zu entdecken.

Da in vielen Fällen die Anwendung eines einzelnen integritätsbasierten Matching-Verfahrens nur unzureichenden Aufschluss auf Korrespondenzen der Schemat liefert, werden häufig verschiedene Matching-Verfahren miteinander kombiniert, um bessere Resultate und eine Reduktion der Vergleiche zu erhalten. Dieser Ansatz wird kombiniertes Matching-Verfahren genannt. Es gibt Hybride Verfahren, die mehrere Verfahren integrieren und zusammengesetzte Matching-Verfahren, welche die einzelnen Verfahren separat durchführen und erst zum Schluss die Ergebnisse kombinieren und auswerten. Die Auswahl und die Anwendungsreihenfolge der einzelnen Matching-Verfahren kann dabei *automatisch* durch Implementierung oder *manuell* durch einen Nutzer durchgeführt werden. Doch in jedem Fall ist abschließend das Eingreifen eines Nutzers notwendig. Matching-Algorithmen geben lediglich Matching-Vorschläge, die von einem Nutzer akzeptiert oder abgelehnt werden müssen.

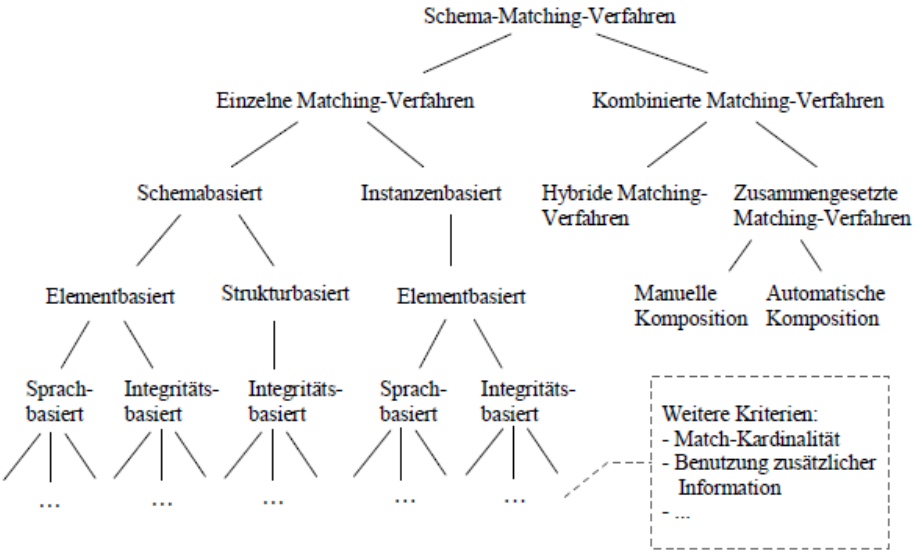


Abbildung 3.7: Baumdiagramm der Schema-Matching-Verfahren [Schna04, S.11]

3.2.2 Umgang mit Datenfehlern

Einer der wichtigsten Punkte beim Umgang mit integrierten Datenquellen, die zuvor heterogen und verteilt vorlagen, ist die Überprüfung der Fehlerbehaftung. Fehler können, wie bereits zuvor erwähnt, die Analysierbarkeit der Daten stark einschränken und zu fehlerhaften Ergebnissen führen, die letztlich falsche Entscheidungen oder Rückschüsse zulassen. Der richtige Umgang mit Daten unbekannter Qualität ist daher äußerst wichtig.

Das Datenmanagement soll in diesem Zuge Fehler entdecken und diese bereinigen können. Dieser Prozess wird in drei Teilprozesse unterschieden: dem Profiling, Assessment und Monitoring. Beim Profiling untersuchen Domänenexperten den Datenbestand mithilfe von Werkzeugen. Hilfsmittel bieten dabei Statistiken wie Maxima und Minima oder Häufigkeitsverteilungen von Attributwerten oder Null-Werten. Ein wichtiges Werkzeug stellt beim Porfiling zudem die Mustererkennung dar. Mithilfe von Mustererkennung können wiederkehrende Anordnungen oder Regelmäßigkeiten in Daten analysiert werden. So lassen sich z.B. Fehler in Anordnungen wie Telefonnummern erkennen, da diese immer gleich typische Muster wie (+**/***/*******, +***-*******…) haben. [LeNa07, S.318]

Der zweite Schritt wird durch das Assessment repräsentiert. Im Assessment werden Bedingungen und Regeln definiert, die die Daten erfüllen sollen. Diese Restriktionen (constraints) können verschiedene Regeltypen haben:

1. Einfach
2. Logisch
3. Probabilistisch
4. Arithmetisch
5. Statistisch

Solche Bedingungen werden direkt von Experten vorgegeben. Der Regeltyp „einfach“ gibt z.B. vor, Datentyp(Lagerbestand) = „numerisch“, also dass der Lagerbestand immer nur in Zahlenwerten angegeben wird. Doch auch Zusammenhänge können realisiert werden.

So kann durch einen Regeltypen „arithmetisch“ erzwungen werden, dass sich der Endeinkaufspreis eines Produktes aus (Nettopreis + Mehrwertsteuer) * Stückzahl + Frachtkosten zusammensetzt. Das Ergebnis des Assessments ist letztlich ein Bericht mit der Anzahl und der Verteilung möglicher Fehler im Datenbestand. [Mül13, S.43; LeNa07, S.325; RaDo00, S.7]

Das Monitoring überwacht schließlich die eingeleiteten Maßnahmen zur Fehlerbereinigung oder zur Beseitigung von Fehlerquellen. Das Reinigen der fehlerbehafteten Daten wird als „data cleaning“, „data cleansing“ oder „data scrubbing“ bezeichnet. Dieser Prozess wird wiederum in zwei Unterprozesse unterteilt. Im ersten Teil werden einfache Fehler, die nur einzelne Datensätze betreffen, bereinigt. In der zweiten Phase werden dann tupelübergreifende Fehler betrachtet und beseitigt. Im Folgenden wird ein typisches Vorgehen, das Müller (2013) beschrieben hat, detailliert nachvollzogen. Dabei handelt es sich nur um ein mögliches Vorgehen. Müller erwähnt ausdrücklich, dass es kein standardisiertes Vorgehen zum data cleaning gibt.

Zu Beginn des data cleaning werden alle Datenwerte in standardisierte Formate überführt. Dieser Prozess korrigiert noch keine Fehler, vereinfacht aber die weitere Bearbeitung der Daten und erleichtert die Fehlerkorrektur. Für jedes Attribut wird zu Beginn dieses Arbeitsschrittes ein Format festgelegt. Zur besseren Vergleichbarkeit von textuellen Daten werden in vielen Fällen sämtliche Buchstaben durch Großbuchstaben ersetzt. Des Weiteren können textuelle Daten durch automatische Rechtschreibprüfungen von den ersten Datenfehlern befreit werden. Das Entfernen von Stoppwörtern („der“, „es“, „und“ usw.) und dem Zurückführen von Wörtern auf ihre Grundform (stemming) vereinfacht die fortlaufende Arbeit mit textbasierten Attributen zunehmend. Letztlich können allgemeingültige Abkürzungen noch durch ihre volle Schreibweise ersetzt werden. [LeNa07, S.325; RaDo00, S.7]

Kontaktdaten werden ebenso auf standardisierte Formate zurückgeführt. Personennamen und Adressen bestehen dabei meist aus mehreren Bestandteilen. Ein Personenname beispielsweise besteht aus Anrede, Titel, Vorname(n) und Nachname, in die er zerlegt werden kann. Ebenso bestehen Adressdaten aus verschiedenen Bestandteilen. Neben Straße und Hausnummer kommen noch Land, Postleitzahl, Ort und ggf. Postfach hinzu. Diese Bestandteile müssen zunächst aufgeteilt und anschließend wie zuvor beschrieben normalisiert werden, sodass beispielsweise die Abkürzung „Str.“ zu Straße ausgeschrieben wird. [LeNa07, S.325; Nau07, S.30]

Letztlich können für Angaben wie Telefonnummern, Daten oder Geldbeträge Standardformate gewählt werden. So werden Datumsangaben z.B. von 01.01.01 auf 01.01.2001 gewandelt oder Telefonnummern mit Landesvorwahl und lückenloser Schreibweise ergänzt. In der Warenwirtschaft ist ein einheitliches Verständnis von Gewichts- und Größenangaben wichtig. Die Konvertierung von Datenwerten in einheitliche Maßangaben ist für ein einheitliches Verständnis ausschlaggebend. Geldbeträge werden anhand des aktuellen Wechselkurses in gewünschte Währungsbeträge konvertiert. [LeNa07, S.326; Nau07, S.30]

Ein großes Problem in der datenbankengestützten Analyse sind fehlende Werte oder Ausreißer. Fehlende Werte können dabei einzelne Werte sein (Nullwerte), aber auch ganze Tupel, Teilrelationen oder ganze Relationen können fehlen. In jeder Hinsicht sind fehlende Werte für die Aussagekraft der Daten schädlich, wenn die speziellen Informationen dieser Daten benötigt werden. Nullwerte können in der Regel durch manuelle oder automatische Überprüfung der Datenmenge gefunden werden. Lückenhafte Datenwertverteilungen, die auf fehlende Teilrelationen schließen, können durch Profiling-Werkzeuge ermittelt werden. [LeNa07, S.327, PLoS05, S.5]

Fehlende numerische Werte werden in diesem Zuge häufig mit Imputation ergänzt. Imputation kann nicht den tatsächlichen Wert herstellen. Imputation lässt vielmehr durch statistische Analyse anderer Werte einen Schluss auf den ungefähren Datenwert zu. Dies geschieht beispielsweise durch die Berechnung von Durchschnittswerten verwandter Einträge oder durch weitaus komplexere Techniken über Datenbeziehungen. Ein weitaus sicheres Hilfsmittel zur Ergänzung fehlender Werte ist die Verwendung von Referenztabellen. Diese helfen bei der Überprüfung von Adressdaten, Telefonnummern oder Bankverbindungen. Mithilfe meist kostenpflichtiger Referenzlisten der Bundesbank, Post oder von Telekommunikationsunternehmen kann neben dem Ergänzen von fehlenden Werten auch die Konsistenz der anderen Daten geprüft werden. Referenztabellen für Adressdaten enthalten Listen aller Ortsnamen, Postleitzahlen, Straßen und Hausnummern.

Sollten Angaben Konsistenzprobleme aufweisen oder lückenhaft sein, können sie mit Hilfe der Angaben der Referenztabelle behoben bzw. vervollständigt werden. Im Hinblick auf Adressdaten bietet das Geocoding/reverse Geocoding eine Alternative zur Referenzliste. Mithilfe der Nutzung einer Geocoding Web API stellen Unternehmen wie Google ihre geografische Datenbank zur Verfügung, um Rückschlüsse auf Standorte von Kunden oder Unternehmen ziehen zu können. Dazu werden geografische und statistische Daten der Umwelt verwendet, um mangelnde Adressangaben zu kompensieren oder den möglichen Standort einzugrenzen oder gar ermitteln zu können.

Ein ausschlaggebender Punkt ist die Beseitigung möglicher Duplikate, die bei der Zusammenführung heterogener Datensätze auftreten können. Hierfür müssen zwei Aufgaben erfüllt werden, damit Duplikate zuverlässig beseitigt werden. Zunächst müssen Duplikate als solche erkannt werden und im zweiten Schritt für das Zusammenfügen der Mehrfacheinträge Inkonsistenzen zwischen den Duplikaten erkannt und behoben werden. Prinzipiell liegt der Duplikaterkennung ein paarweise durchgeführter Vergleich aller Tupel zu Grunde. Ein Ähnlichkeitsmaß gibt dabei die Übereinstimmung an. Ist das Maß größer als ein bestimmter gewählter Schwellwert, werden die Tupel als mögliche Duplikate gekennzeichnet. Da eine Duplikatsuche sowohl effizient als auch genau sein soll und ein Paarvergleich aller Tupel eine quadratische Anzahl an Tupelvergleichen zur Folge hätte, werden nicht alle Tupel miteinander verglichen. Tuperlvergleiche werden vermieden, indem Tupel anhand von Relationen in Partitionen eingeteilt werden und nur Tupel innerhalb ihrer Partition verglichen werden. Die Effizienz betrifft jedoch nicht nur den Laufzeitaspekt. Auch die Genauigkeit der Duplikaterkennung spielt eine Rolle. Fehler sind unvermeidbar, sodass Duplikate nicht erkannt werden oder nicht-Duplikate fälschlicher Weise als Duplikate erkannt werden. Die Effizienz wird mit zwei Maßen gemessen: *Precision* und *Recall.* Eine hohe Precision wird durch ein strenges Ähnlichkeitsmaß erreicht. Das hat zur Folge, dass gefundene Duplikat mit hoher Wahrscheinlichkeit auch wirklich Duplikate sind. Jedoch werden schnell Duplikate mit nur geringer Übereinstimmung nicht als solche erkannt. Ein hoher Recall hingegen heißt, dass viele der tatsächlichen Duplikate gefunden wurden. Das wird durch eine tolerantes Ähnlichkeitsmaß erwirkt. Der Nachteil dabei ist, dass auch viele nicht-Duplikate als Duplikate erkannt werden. Abbildung 3.8 zeigt den Zusammenhang. [LeNa07, S.325ff; RaDo00, S.8]

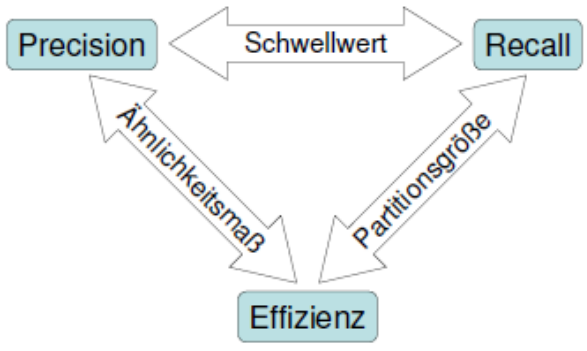


Abbildung 3.8: Zusammenhang Effizienz-Precision-Recall [LeNa07, S.334]

4. Qualitätssicherung von Supply-Chain-Daten

In Kapitel 2 wurden grundlegende Beschaffenheiten von Supply-Chain-Daten und deren Kategorisierung besprochen. Im Anschluss wurde eine gängige Darstellungsform für Supply-Chain-Daten beschrieben und geschildert wie die Datenversorgung dieser Darstellungsform abläuft. Kapitel 3 gab im Folgenden Aufschluss darauf, welche Problematiken bei der Zusammenführung verschiedener Datenquellen zur Datenversorgung von Datenbankensystemen auftreten können und wie Datenheterogenität zwischen Datenquellen und Fehlern in Daten identifiziert und ggf. behoben werden können. Im folgenden Kapitel wird das erlangte Wissen auf den Anwendungsbereich der Supply-Chain projiziert und einzelne Verfahren und Vorgehen auf Anwendbarkeit geprüft. Zu diesem Zweck wird in diesem Kapitel ein Konzept zu Bereinigung von Supply-Chain-Rohdaten erarbeitet. Das Konzept repräsentiert die Schritte der Vorverarbeitung, vor der eigentlichen Integration in das finale Supply-Chain-System. Das Ziel ist, die Daten in ihren ursprünglichen Datenformaten weitestgehend so zu bearbeiten, dass eine problemfreie Integration in das Zielsystem ermöglicht wird.

4.1 Steigerung der Qualität von Supply-Chain-Daten

Das vorangegangene Kapitel hat gezeigt, dass kein Datensatz von Fehlern ausgeschlossen ist und Nachlässigkeit in der Datenhaltung oder menschliches Versagen beim Erstellen von Datensätzen zu Fehlerpotentialen führen. Ebenso wurde erklärt, dass vor der möglichen Integration von Daten Heterogenität zu beachten ist. Aus diesen Gründen ist ein klares strukturiertes Vorgehen nötig, um Supply-Chain-Rohdaten von störenden Faktoren zu bereinigen und sie auf ein möglichst hohes Datenqualitätsniveau zu heben. Anhand der in Abschnitt 2.2.3 gezeigten Datenverhältnisse in heutigen Unternehmen wird hier eine systematische Vorverarbeitung von Excel-Rohdaten vor der Integration in analytische Systeme (hier durch Supply Chains dargestellt) erläutert. Die verwendeten Fehlerdetektions- und Bereinigungskonzepte beziehen sich teilweise auf von Excel bereitgestellte Features.

4.1.1 Konzept zur Fehlerbereinigung von Supply-Chain-Daten

Ziel des hier vorgestellten Konzepts ist es Rohdaten weitestgehend so vor zu verarbeiten, dass eine möglichst hohe Datenqualität bei der Datenintegration in eine Supply Chain vorliegt. Das hier in Abbildung 4.1 vorgestellte Vorverarbeitungskonzept wird durch ein Verlaufsdiagramm dargestellt und zeigt die Hauptpunkte der grundlegenden Schritte zur Datenaufbereitung der Rohdaten. Teilkonzepte stützen sich dabei auf bereits bekannte Verfahren der Aufbereitung von Daten zur Nutzung in Datenbanken. Zudem versucht das vorgestellte Vorgehen einen klar strukturierten Arbeitsverlauf von Rohdaten mit unbekanntem Format und Datenqualitätsmaß zu integrationsfähigen Eingangsdaten, die ein zielsystemkonformes Format aufweisen, vorzugeben.

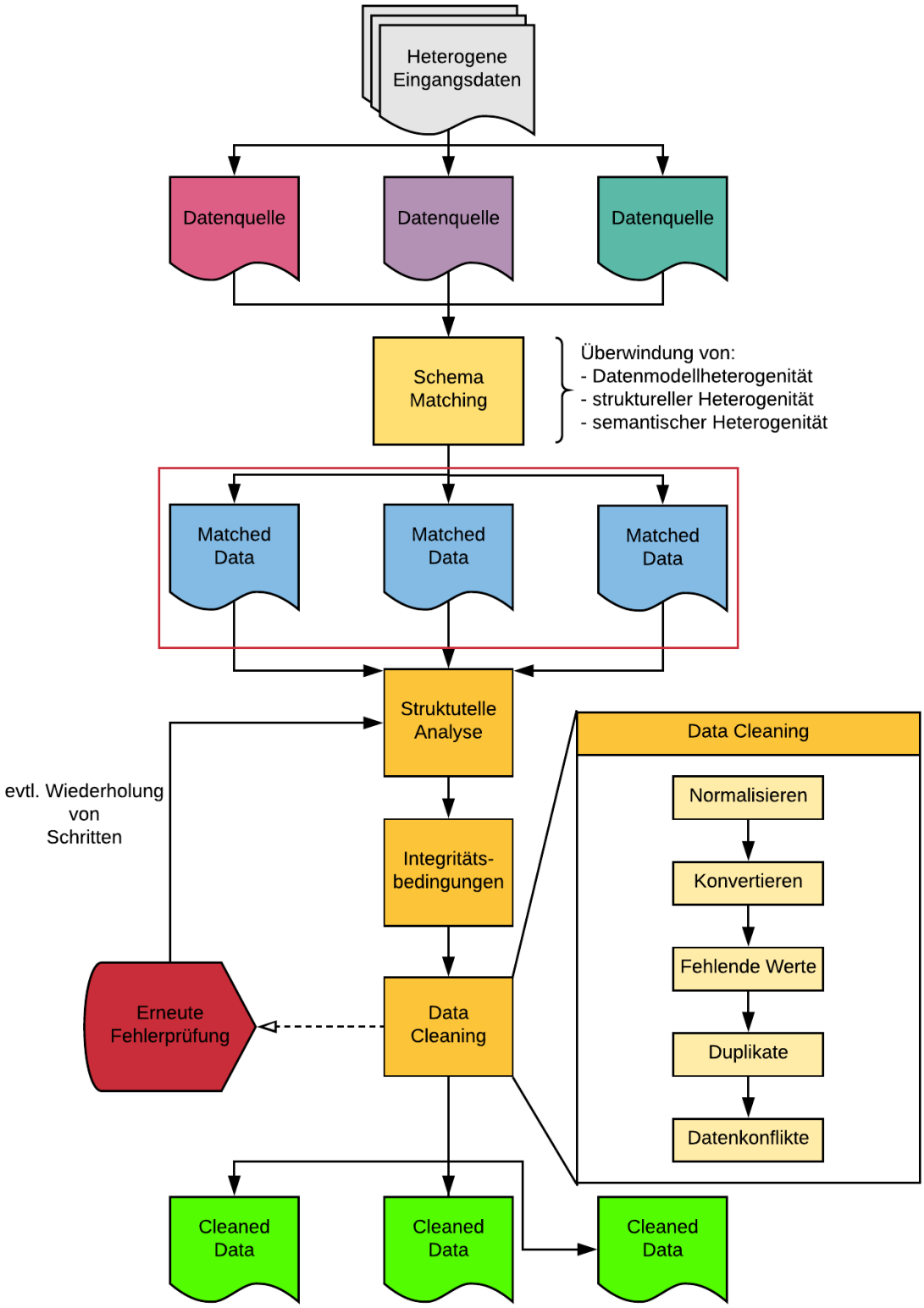


Abbildung 4.1: Ablaufdiagramm der Aufbereitung von Supply-Chain-Rohdaten

In der Verlaufsdarstellung lässt sich das Vorgehen durch vier Arbeitsprozesse zur Homogenisierung und Fehlerkorrektur von Rohdatensätzen beschreiben. Diese vier Arbeitsprozesse gruppieren sich in zwei grobe Teilprozesse. Als erstes der Beseitigung von Heterogenität, damit die Daten im Anschluss in einheitlicher Form vorliegen, dadurch lassen sich Fehlerkorrekturprozesse einfacher auf die Daten anwenden. Zudem lassen sich die Daten nur in angepasster Form in das Zielsystem integrieren. Der zweite Teilprozess, der die Arbeitsprozesse „Strukturelle Analyse“, „Integritätsbedingungen“ und „Data Cleaning“ enthält, dient zur Fehlerfindung und Beseitigung, damit die Eingangsdaten, die bis dahin in unbekannter Qualität vorliegen, in eine möglichst ideal nutzbare Datenqualität gebracht werden. Die exakten Schritte des Konzeptes lassen sich wie folgt beschreiben:

Schema-Matching

Zunächst werden die Daten auf ihren Heterogenitätsgrad untersucht und mittels verschiedener Verfahren von auftretenden Arten von Datenmodellheterogenität, struktureller Heterogenität und semantischer Heterogenität befreit. Datenmodellheterogenität tritt nur auf, wenn die zu integrierenden Daten unterschiedlichen Dateiformaten unterliegen. Um Heterogenität festzustellen, müssen die Schemata der Rohdaten untereinander, als auch mit dem Zielschema der bereits integrierten Supply-Chain-Daten, verglichen werden. Zunächst wird die vorliegende Form der Daten überprüft. Dazu wird die verwendete Semantik der Datensätze abgeglichen und, falls notwendig, mittels Überführungsfunktion in die Semantik des Zielzeichensatzes konvertiert. Hierfür können Überführungstabellen verwendet werden um z.B. kyrillische oder griechische Zeichensätze in lateinische Zeichensätze zu überführen. Ein Beispiel für eine Zeichenüberführungstabelle (zeigt Tabelle 4.1).

Griechischer Zeichensatz	Lateinischer Zeichensatz
Α, α	A, a
Β, β, ϐ	B, b
Γ, γ	G, g
Δ, δ	D, d
Ε, ε, ϵ	E, e
Ζ, ζ	Z, z
Η, η	I, i
Θ, θ, ϑ	TH, th
Ι, ι	I, i
Κ, κ, ϰ	K, k
Λ, λ	L, l
Μ, μ	M, m
Ν, ν	N, n
Ξ, ξ	X, x
Ο, ο	O, o
Π, π, ϖ	P, p
Ρ, ρ, ϱ	R, r
Σ, σ, ς, c	S, s
Τ, τ	T, t
Υ, υ	Y, y
Φ, ϕ, φ	F, f
Χ, χ	CH, ch
Ψ, ψ	PS, ps
Ω, ω	O, o

Tabelle 4.1: Überführungstabelle Zeichensatz „Griechisch-Lateinisch“

Datenverarbeitungsprogramme wie zum Beispiel Microsoft Excel bieten die Möglichkeit Visual Basic basierte Module zu definiere, mit deren Hilfe solche Prozesse durchgeführt werden können. Die in Visual Basic definierte Funktion „Replace“ kann eine beliebige Zeichenkette durch eine andere ersetzen. Dieser Vorgang lässt sich beliebig oft ineinander verschachteln. Somit können direkte Überführungen für den gesamten Datensatz schnell und unkompliziert durchgeführt werden.

Durch Begutachten der Schemata der einzelnen Datensätze können innerhalb der Überwindung semantischer Heterogenität unbekannte Bezeichner der Attribute erkannt werden. Dazu muss überprüft werden, ob sich anhand des Attributnamens eine Funktion ableiten lässt. Wenn sich keine ursprüngliche Funktion des Datenattributs erschließen lässt, wird das Attribut aus dem Datensatz entfernt, da ohne Funktion kein Nutzen aus den zusätzlichen Daten gezogen werden kann. Vergleiche über den dargestellten Datentypen lassen in Kombination mit der entsprechenden Attributbezeichnung Rückschlüsse auf Homonyme und Synonyme zu. Zu diesem Zweck müssen die vorliegenden Datenquellen abgeglichen werden und Bezeichner der Attribute an Zielsystem-interne Bezeichnungen angepasst werden. Einfache Namenskonflikte wie Homonyme oder Synonyme bzw. eine Nicht-Zuordenbarkeit von Attributfunktionen lassen sich am einfachsten händisch korrigieren.

An dieser Stelle liegen die Rohdaten nun in gematchter Form vor. D.h. es sollten keine Heterogenität mehr zwischen den Rohdaten-Schemata mehr vorhanden sein. Ausgehend von dieser Basis kann mit der Bereinigung der Datenfehler fortgefahren werden. Die Datenfehlerbereinigung wird in drei Abschnitte unterteilt.

Strukturelle Analyse

Zunächst werden die Daten strukturell untersucht. Innerhalb dieses ersten Fehleranalyseschritts werden Null-Werte gesucht und markiert. Um leere Zellen zu identifizieren bietet Excel Funktionen um Leerzellen zu markieren. Mithilfe der „bedingten Formatierung“ können Regeln zum Hervorheben von Zellen definiert werden, mit dessen Hilfe auch Leerzellen gekennzeichnet werden können. Mit der Regel „=ISTLEER(*Startzelle*)“ lassen sich alle Null-Werte innerhalb des Dokuments markieren. Welche Auswirkungen fehlende Informationen in Datensätze für die Auswertbarkeit haben, wurde in Abschnitt 3.1.2 erläutert. Die markierten Werte können im Fortschreiten der Fehlerkorrektur gefüllt oder entfernt werden. Anschließend können numerische Werte, die dem Format des repräsentierten Attributs nicht entsprechen, aussortiert bzw. markiert werden. Selbst wenn numerische Werte von Telefonnummern, EANs oder Kontonummern nicht leer sind, so können sie unvollständig oder falsch sein. Auch ohne direkten Abgleich durch Referenzlisten lassen sich bereits im Vorfeld die Werte als fehlerhaft markieren, die zu wenig, zu viele oder falsche Ziffern besitzen oder ggf. gar keine Zahlenwerte sind. In Excel lassen sich auch zu diesem Zweck überprüfende Funktionen nutzen, die Anzeigen, ob es sich um Zahlen handelt und die gewünschte bzw. notwendige Zeichenzahl eingehalten wird.

Integritätsbedingungen

Im Anschluss können für die Dateneinträge der Datensätze Integritätsbedingungen definiert werden. Mit Hilfe dieser Regeln werden unlogische Werte und Dummy-Werte im Datensatz lokalisiert. Darunter fallen insbesondere Strukturdaten wie Größen- oder Gewichtsangaben zu Produkten oder Angaben zu Liefermengen. Die Regeln müssen so gewählt werden, dass sie die realen Gegebenheiten der Supply-Chain-Daten möglichst gut umreißen. Dafür ist eine Kenntnis über Realwerte nötig. Je genauer die Regeln Werte eingrenzen, desto größer ist die Wahrscheinlichkeit, dass Ausreißerwerte aufgefunden werden. Schlecht gewählte Regeln führen dazu, dass Ausreißerwerte in der erwarteten Realwertmenge verbleiben oder Realwerte als Außerreißerwerte deklariert werden. Für numerische Werte können Gültigkeitsbereiche festgelegt werden, sodass Werte immer positiv oder größer Wert „x“ sein müssen.

Aber auch unrealistische Abweichungen können direkt einbezogen werden. Angaben wie beispielsweise „maximale Ladungsmenge“ von Transportfahrzeugen können auf eine realistische Obergrenze beschränkt werden. Das Abfangen solcher abweichenden Werte hilft spätere, aus Analysen hervorgehende, Ergebnisse zu verbessern. Außreißerwerte können Auswertungen über Datenmengen verfälschen. Ein Beispiel für das verfälschende Potential von genannten Abweichungen zeigt Tabelle 4.2

FahrzeugID	Fahrzeugtyp	Netto Gewicht (kg)	Ladungsvolumen (m^3)	Ladungsgewicht (kg)
00001	LKW	20.000	75	21.500
00002	LKW	25.000	80	21.000
00003	Kleintransporter	3.000	10	1.600
00005	LKW	20.000	75	20.000
00009	Kleintransporter	2.500	10	15.000 (1.500 real)

Tabelle 4.2: Beispiel für Ausreißerwert

Im obigen Beispiel ist ein Außreißerwert im Attribut des maximalen Ladungsgewichtes vorhanden. Schon bei simplen Auswertungen des Mittelwerts der maximalen Beladung von LKWs und Kleintransportern führt dies zu falschen Ergebnissen. Die mittlere Beladungsgrenze der LKWs liegt korrekterweise bei 20833 kg – die der Kleintransporter hingegen bei 8300 kg. Der Realwert liegt für die Kleintransporter in diesem Beispiel bei 1550 kg. Ausreißerwerte können demnach erhebliche Veränderungen an Auswertungsergebnissen verursachen. In größeren Datensätzen kommt es auf die Menge und die Schwere der Fehler an, wie stark diese bei Auswertungen Ergebnisse beeinflussen. Auch die Art der Abfrage ist entscheidender Faktor für das Fehlergewicht. In großen Datensätzen werden Einzelfehler in Mittelwert oder Median Berechnungen wahrscheinlich nur wenig Einfluss auf das Endergebnis haben, wogegen Abfragen nach Maxima oder Minima komplett verfälscht werden können.

Da Microsoft Excel Vergleichsoperatoren bereitstellt, ist die Einschränkung von Werten in Gültigkeitsbereiche problemlos umsetzbar. Für einzelne Attribute der Datensätze können obere und untere Grenzen festgelegt werden und mittels der „bedingten Formatierung“ Zahlenwerte außerhalb der gesetzten Grenzwerte für die Weiterverarbeitung markiert werden. Das weitere Vorgehen hängt anschließend von der Beschaffenheit des Fehlers ab.

Anhand der gekennzeichneten Bereiche kann direkt eingeschätzt werden, wie stark die vorliegende Datenquelle von Datenfehlern betroffen ist und welche Schritte der Datenbereinigung genutzt werden müssen, um die Datenfehler bestmöglich zu kompensieren.

Data Cleaning

In dem hier als „Data Cleaning“ bezeichneten Teilprozess werden die zuvor identifizierten Dateninkonsistenzen bearbeitet. Das Vorgehen der Datenbereinigung ist wiederrum in fünf Arbeitsschritte unterteilt, die unterschiedliche Aufgaben erfüllen. Das Vorgehen ist dabei an das in Abschnitt 3.2.2 beschriebene Verfahren von Felix Naumann und Ulf Leser angelehnt. Zu diesem Zweck werden die Datensätze zunächst in den Schritten „Normalisieren“ und „Konvertieren“ harmonisiert und in bearbeitungstaugliches Format gebracht. Anschließend werden vorkommende Fehler in den Schritten „Fehlende Werte“, „Duplikate“ und „Datenkonflikte“ bereinigt. Je nach Datentyp und Datenkategorie weisen die Unterpunkte unterschiedliche Verfahren zur Datenbereinigung auf.

Normalisieren

Während der Normalisierung werden die Daten zur vereinfachten Fehlerüberprüfung in standardisierte Formate überführt. Textuelle Dateneinträge werden dazu zunächst komplett in Großbuchstaben überführt. Dazu kann die selbe Überführungsfunktion genutzt werden, die bereits für die Zeichenüberführung von Alphabeten genutzt wurde.

Stammdaten durchlaufen zusätzliche formatierende Schritte. Personennamen, von z.B. Ansprechpartnern, werde in ihre Bestandteile (z.B. Anrede, Titel, Vorname(n), Nachname) zerlegt und sortiert. Ebenso werden Adressdaten zerlegt und sortiert (z.B. Straße, Hausnummer, PLZ, Ort, Land). Liegen die Stammdaten in einheitlicher Reihenfolge vor, ist das Zerlegen der Daten in Einzelattribute mithilfe der „Text-in-Spalten"-Funktion von Excel möglich. Dadurch werden zusammenhängende Adresszeilen, die zuvor in einem Attribut gehalten wurden, in gewünschte Attributmenge aufgeteilt.

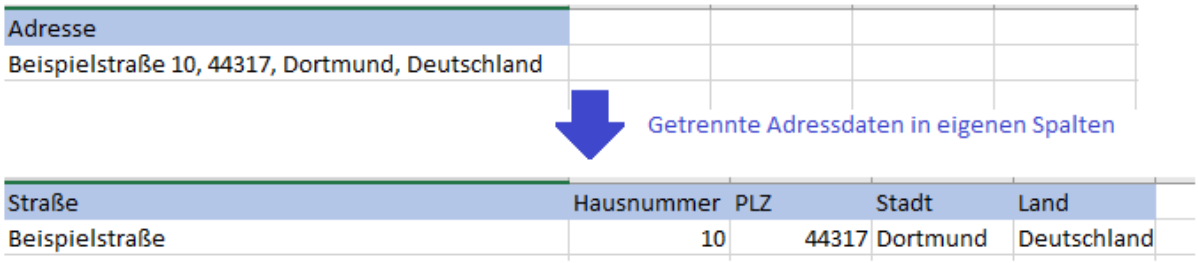


Abbildung 4.2: Formatierung von Adressdaten mit Excel Funktion „Text-in-Spalten"

Wenn Stammdaten unformatiert vorliegen, z.B. durch unregelmäßige Darstellung der Adresszeile oder unterschiedliche Informationsmenge, muss versucht werden, die Angaben zunächst in ein einheitliches Erscheinungsbild zu bringen. Ein entsprechend angepasstes Visual Basic Modul kann eine entsprechende Umformatierung vornehmen. Das Modul (Anhang I) überprüft dazu zunächst die ausgewählten Zeilen auf Trennung von Zahlen und Buchstaben. Anschließend wird der Text anhand von Leerzeichen in seine Bestandteile zerlegt und die Wörter danach in einmaliger Ausführung wieder in der richtigen Reihenfolge zusammengesetzt. Postleitzahlen und Telefonnummern werden während des Prozesses aussortiert, ebenso wie die möglichen verwendeten Abkürzungen wie „PLZ:". Durch die integrierte Trim-Funktion werden überflüssige Leerzeichen entfernt. Dieses Vorgehen verbessert die textuelle Darstellung und unterstützt damit spätere Prozesse während Wortfindungs- oder Vergleichsoperationen, wie sie bei der Duplikatsuche oder dem Geocoding Anwendung finden.

Weitere Stammdaten die sich in standardisierte Formate bringen lassen, sind Telefonnummern, EANs, Geokoordinaten und Datumsangaben. Je nach Darstellung des Zielschemas lassen sich diese Stammdaten in ein gewünschtes Zielformat überführen z.B. durch das Ergänzen der angegebenen Telefonnummer mit einer Landeskennung oder das Entfernen von Leerzeichen und Bindestrichen. Ein einfaches Beispiel dafür stellen Telefonnummern dar. Auch hier können wieder Visual Basic Module verwendet werden, um eine einheitliche Darstellungsform zu generieren. Ein Beispiel für die Normalisierung von Telefonnummern zeigt Abbildung 4.3.

```
Sub Telefonnummern()
Dim Nummer As Range

Selection.NumberFormat = "@"
For Each Nummer In Selection
Nummer = "+49" & Replace(Replace(Replace(Replace(Replace(Replace _
(Replace("+" & Nummer, "+49", ""), "+0", ""), "+", ""), "-", ""), "/", ""), _
" ", ""), ")", ""), "(", "")
Next
End Sub
```

Abbildung 4.3: Visual Basic Code für VBR-Editor Modul zur Normalisierung von Telefonnummern

Konvertieren

Mit Hilfe von Konvertierungsfunktionen können numerische Werte von der vorliegenden Einheit in eine gewünschte Zieleinheit umgerechnet werden. Das betrifft vor allem Gewichts-, Längen- und Währungseinheiten. Gerade bei Stammdaten von Produktlisten ist es wichtig, eine einheitliche Einheit der jeweiligen Attribute zu wählen, um Vergleichbarkeit zu gewährleisten. Eine Möglichkeit bietet hierzu Excel mit dem Befehlt „UMWANDELN“. Excel hat bereits eine große Vielzahl von Umrechnungsfaktoren von Maßeinheiten implementiert. Dies deckt unter anderem Einheiten für Gewichte, Entfernungen, Massen, Temperaturen oder Energie ab.

Währungseinheiten für Produktpreise oder in Transaktionsbelegen werden anhand des aktuellen Wechselkurses von der Ursprungswährung in die Zielwährung umgerechnet. Das heißt, vor der Umrechnung muss der jeweilige Wechselkurs zwischen der ausgehenden Einheit und der Zielwährung erfasst werden. Teilweise ist es nötig, den ursprünglichen Preis ebenfalls zu speichern, da sich bei veränderndem Wechselkurs auch der Preis ändert (z.B. in Produktlisten für den Einkauf).

Fehlende Werte

Im Zuge der Überprüfung fehlender Werte und von Ausreißern wird ein Großteil der Dateninkonsistenzen bearbeitet. Zum einen verfälschen fehlende Werte in Transaktionsdaten Aggregationsanfragen z.B. an Produktinformationen, wenn diese unvollständig sind. Zum anderen können fehlende Werte in Stammdaten wie Adressen zu Fehlern führen, sobald die Informationen zur Planung herangezogen werden sollen. Genau aus diesen Gründen müssen fehlende Werte schon vor der Integration der Rohdaten in die Supply Chain gefunden und bestenfalls ergänzt werden. Je nach Vorkommen und Datentyp können verschieden Herangehensweisen für die Fehlerkorrektur in Betracht gezogen werden. Speziell bei fehlenden Werten von Kunden- oder Lieferantendaten bestehen verschieden Herangehensweisen, je nach fehlendem Datentyp und Fehlerumfang. Wie bereits in 3.2.2 erwähnt, bieten Referenztabellen oftmals Abhilfe, fehlende Stammdatenwerte zu ergänzen. Fehlende Einträge wie der Straße, Hausnummer oder Postleitzahl können mithilfe von Postleitzahlentabellen oder Branchenlisten abgeglichen werden. So kann auch die Konsistenz vollständiger Datensätze geprüft werden. Der Abgleich scheitert, sobald zu wenig Informationen vorhanden sind, um anhand von Listen Referenzen zu den verbleibenden Daten zu erstellen. Sobald kein genauer Sachverhalt mehr erstellt werden kann, können nur Schätzwerte herangezogen oder erstellt werden. Dies erzeugt zwar keine tatsächlich gültigen Werte, kann aber zu einer akzeptablen Näherung führen, die die Nutzung der Daten ermöglicht. Ein anschauliches und oftmals nötiges Beispiel ist die Bestimmung eines Standpunktes. Ausgegangen von dem Szenario, dass der in den Datensätzen angegebene Standort eines Lieferanten nur durch seine Postleitzahl angegeben ist und weitere Adressinformationen fehlen, jedoch eine genaue Angabe durch Geokoordinaten erwünscht ist. Eine genaue Bestimmung des Standortes ist in diesem Punkt nur per Näherungswert zu erreichen. Als Beispiel wird ein genauer Standort in dem Postleitzahlenbereich „44309“ gesucht.

Um nun einen zufälligen Punkt in diesem Bereich zu bestimmen, wird mithilfe der nördlichsten, südlichsten, westlichsten und östlichsten Koordinate des Bereichs ein Rechteck über dem PLZ-Gebiet aufgespannt. Die hier gewählten Punkte sind (Angabe in Grad und Dezimalgrad):

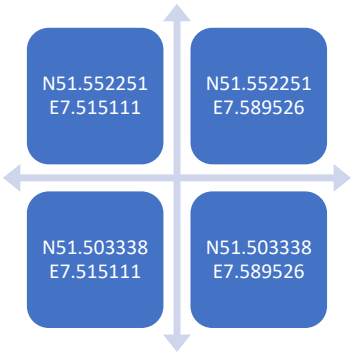


Abbildung 4.4: Gebietsrandpunkte zur Bestimmung von Zufallskoordinaten

Anhand der Dezimalgrade lassen sich hier nun Zahlenbereiche festlegen. In diesem Beispiel für die Vertikale (503338-552251) und in der Horizontalen (515111-589526). Innerhalb dieser beiden Zahlenbereiche müssen im Anschluss die gleiche Anzahl Zufallszahlen erzeugt werden und zu Tupelpaaren kombiniert werden. Der aufgespannte Bereich wird in Abb. 4.5 und das Resultat in Abb. 4.6, indem 100 Zufallspunkte in dem aufgespannten Gebiet platziert wurden, dargestellt. Im Anschluss muss für jeden gesetzten Punkt überprüft werden, ob dieser auch in der gesuchten Postleitzahl liegt.

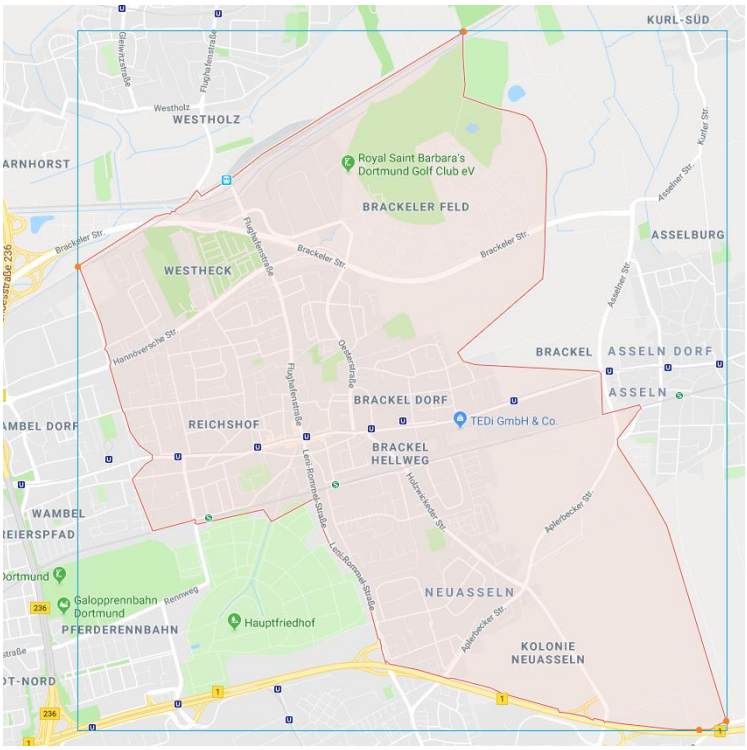


Abbildung 4.5: Postleitzahlenbereich 44309 (rot) – Durch Maximalkoordinate Nord-Süd-Ost-West aufgespanntes Rechteck (blau), [GoogleMaps]

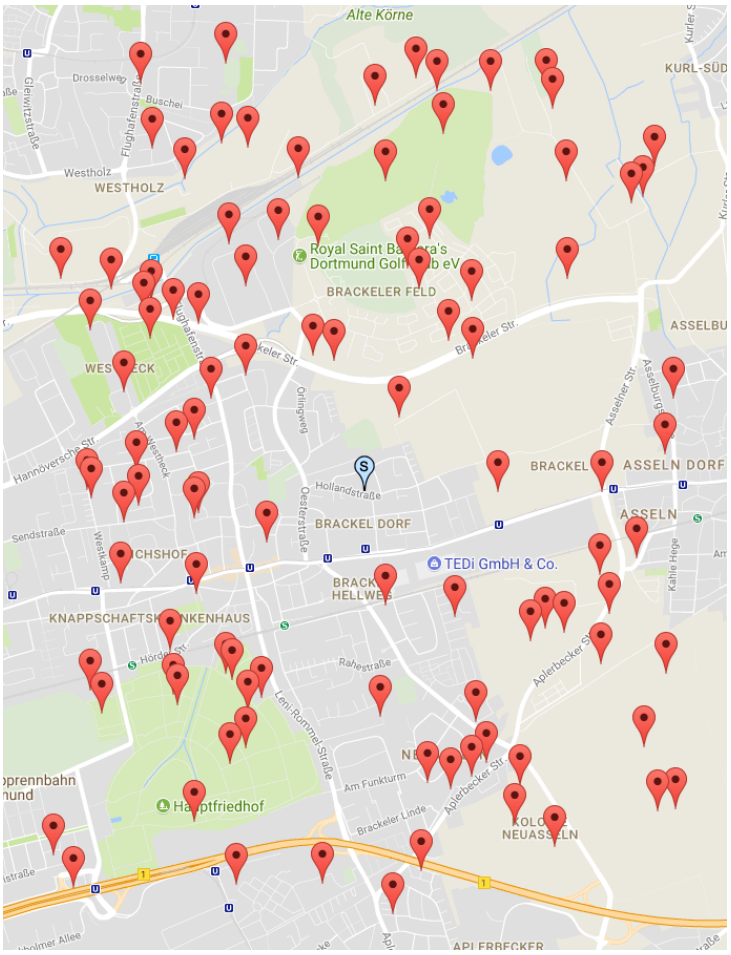


Abbildung 4.6: 100 Random Points im zuvor aufgespannten Gebiet, [geomidpoint.com]

Aus der Summe von „korrekten“ Punkten lässt sich ein beliebiger Punkt auswählen, der als Näherung des tatsächlichen Standpunktes fungiert. Mit dieser Methode werden zwar nicht die tatsächlichen Adressdaten rekonstruiert, jedoch werden Analyse- und Planungsprozessen ein Näherungswert der Geokoordinaten bereitgestellt.

Zulieferer	Straße	Nr	PLZ	Stadt	Land	Geokoordinaten	Entfernung (km)
LagerA	Heeper Straße	130	33607	Bielefeld	Deutschland	52.022415, 8.556056	158
LagerB			59494		Deutschland		
LagerC	Leuschner-straße	97	34134	Kassel	Deutschland	51.289761, 9.450173	116

Tabelle 4.3: Beispiel für lückenhafte Adressdaten

Das Beispiel aus Tabelle 4.3 zeigt verschiedene Standorte, die hier als Großlager angenommen werden. Es soll anhand der Entfernung entschieden werden, aus welchem der Großlager, das fiktiv in Dortmund ansässige Unternehmen, Bauteile für die Produktion bezieht. „LagerB“ weist einen unvollständigen Adresssatz auf und anhand der Geokoordinaten kann keine Entfernung bestimmt werden, sodass die Entscheidung für „LagerC“ getroffen wird. Wenn mithilfe der Postleitzahl „59494“ eine Koordinate approximiert wird, ergibt sich je nach bestimmter Zufallskoordinate im Postleitzahlenbereich eine Entfernung zwischen 50 – 60 (km). Ausgehend von diesem bestimmten Wert, würde die Entscheidung für „LagerB“ fallen. Das zeigt, dass approximierte Werte zwar keine Realwerte rekonstruieren können, jedoch zur Entscheidungsfindung oder Auswertung durchaus herangezogen werden und brauchbare Näherungen darstellen.

Sind komplette Adressdaten vorhanden und eine Angabe in Geokoordinaten ist erwünscht, können mithilfe von Geocoding entsprechende Geokoordinaten zum Standort erstellt werden. Reverse Geocoding kann herangezogen werden, wenn Geokoordinaten vorhanden sind, der übrige Adressdatensatz jedoch unvollständig ist. Unternehmen wie Google oder Microsoft stellen zu diesem Zweck ihre geografischen APIs zur Verfügung, mit deren Hilfe sich Adressdaten in Geokoordinaten und Geokoordinaten in Adressdaten umrechnen lassen. Mithilfe von Visual Basic Modulen lässt sich die vorhandene Adresszeile mit der Datenbank der geografischen API abgleichen und fehlende Geokoordinaten werden nach Möglichkeit direkt ergänzt.

Fehlende Einträge in Datensätzen wie Transaktionsaktivitätsdaten oder Transaktionskontrolldaten lassen sich meist gar nicht vollständig rekonstruieren. Fehlende Zeitstempel getätigter Transaktionen oder fehlende Angaben zu Ladungsmengen oder Gebindegrößen haben keine Korrelationen zu anderen Werten und können nur mittels Imputation gefüllt werden. Auch hier werden keine Realwerte erzeugt, jedoch mittels statistischer Analyse beispielsweise ein Mittelwert ermittelt. Ein Beispiel dafür stellt eine fehlende Lieferzeit einer Transaktion dar. Da sich die Lieferzeit nicht ermitteln lässt (sofern keine Werte vorhanden sind, aus denen sie sich berechnen lässt), kann diese nur geschätzt werden. Eine nützliche Schätzung ist die Erhebung eines Mittelwertes aus den Lieferzeiten zwischen identischen Lieferpunkten.

Ausreißer-Werte werden ähnlich behandelt wie fehlende Werte. Ausreißer sind meist unrealistisch hohe oder niedrige numerische Werte, die festgelegte Regeln verletzen. Handelt es sich um Tippfehler, die jedoch Experten Rückschuss auf den Realwert lassen, so kann dieser händisch korrigiert werden (vgl. Tabelle 4.2). Nicht nachvollziehbare Werte werden wie Null-Werte behandelt und können nur abgeschätzt werden.

Ein besonders schwerwiegender Fehlerfall ist, wenn Datenfehler in hierarchischen Daten oder in Schlüsseln vorliegen. Da diese Daten nur in exakter Form nutzbar sind und somit Näherungen keine Funktion erfüllen, können fehlende Werte oder Datenfehler nicht korrigiert werden. Die davon betroffenen Daten sind meist nutzlos, da keine eindeutige Zuordnung des beschriebenen Objektes mehr möglich ist.

Duplikate

Im nächsten Schritt werden Duplikate behandelt. Duplikate treten relativ häufig bei der Zusammenführung mehrerer Datenquellen auf. Gerade bei der Zusammenführung von internen Datenquellen werden Einträge oftmals mehrfach geführt. Sowohl zwischen den einzelnen Datenquellen, als auch in derselben Datenquelle, können Dubletten auftauchen. Durch den Vergleich der einzelnen Einträge lassen sich mithilfe eines Gleichheitsmaßes Duplikate erkennen, um diese im Anschluss zusammenzuführen. Als besonders gutes Vergleichskriterium eignen sich Schlüssel bzw. Metadaten. Ist eine eindeutige Kennung mehrfach vorhanden, kann davon ausgegangen werden, dass es sich um ein Duplikat handelt. Das Beispiel in Abbildung 4.7 zeigt das Ergebnis einer Duplikatprüfung nach einer Kennung. In diesem Fall wurde anhand des Attributs „KundenID“ ein doppelt geführter Eintrag gefunden. Dazu wurde eine Formatierungsregel mit der Funktion „ZÄHLENWENNS()“ definiert, die zählt, wie oft Zeilen dasselbe Attribut aufweisen. Alle Zeilen deren Attribut mehr als einmal vorkam, werden durch die Formatierungsregel markiert.

1	KundenID	Name	Adresse	PLZ	Land	Beschreibung
2	K14572	Kunde 1	Westenhellweg 114	44137	Deutschland	Store
3	K16743	Kunde 2	Rheinische Straße 5	44137	Deutschland	Store
4	K23654	Kunde 3	Dorstfelder Hellweg 9	44149	Deutschland	Store
5	K14572	Kunde 4	Westenhellweg 114	44137	Deutschland	Store
6	K54872	Kunde 5	Mecklenburger Straße 21	58708	Deutschland	Store

Abbildung 4.7: Duplikatprüfung nach Schlüselkennung

Doch nicht immer lassen sich Duplikate durch einen Vergleich eines einzelnen Attributs identifizieren. Viele Duplikate sind nur teilidentisch, beschreiben jedoch den selben Realweltbezug. So können Datensätze identisch sein, jedoch trotzdem mit unterschiedlichen Schlüsselwerten geführt werden oder bestimmte Werte fehlen in einem oder mehreren Versionen des gleichen Dateneintrages. Kleine Abweichungen oder andere Schreibweisen können immer noch das selbe Objekt oder denselben Sachverhalt beschreiben, jedoch werden sie nicht mehr von der einfachen Duplikatsprüfung als Duplikat erkannt. Eine Möglichkeit trotz dieses Problems weitere Duplikate auszumachen, ist eine Duplikatsprüfung immer nach mehreren Attributen gleichzeitig auszurichten. Unformatierte Attribute oder stark variable Werte wie Messwerte oder Geokoordinaten eignen sich nur sehr bedingt für die Duplikatsprüfung. Stattdessen sollten fixe Werte wie Strukturdaten zur weiteren Prüfung miteinbezogen werden. Eine weitere Möglichkeit teilidentische Attribute zu lokalisieren, ist zu prüfen, ob ein Attribut Teilrelation eines anderen ist. Dadurch kann kontrolliert werden ob Einträge vorhanden sind, die identisch sind und lediglich zusätzliche Zeichen enthalten und deshalb nicht von der Duplikatsuche erkannt wurden.

Datenkonflikte

Duplikate die in einem oder mehreren Attributen voneinander abweichen, werden als Datenkonflikte bezeichnet. Im letzten Schritt der Datenbereinigung müssen diese Konflikte beseitigt werden, um eine Zusammenführung der Duplikate zu bewerkstelligen.

Ist Datensatz A ein Duplikat von B wird unterschieden, ob A und B sich ergänzen (A oder B hat eine oder mehrere zusätzliche Informationen) oder ob A und B sich widersprechen (A oder B hat ein oder mehrere Attribute mit sich unterscheidenden Werten). Ergänzt ein Dateneintrag den Anderen zum Beispiel durch Führung eines Produktes in Produktelisten aus der Fertigung und der Lagerhaltung, so wird der Datensatz zu einem zusammengeführt und wird durch das zusätzliche Attribut ergänzt. Bei Widerspruch zwischen den Duplikaten muss geklärt werden, welcher der Werte, in dem sich die Datensätze widersprechen, fehlerhaft ist. Der Datensatz mit dem fehlerhaften Eintrag wird aus der Relation entfernt. Kann keine Funktion in dem sich widersprechenden Attribut nachvollzogen werden, können die Datensätze ebenfalls als gleich betrachtet werden.

Eine erneute Überprüfung der Datenqualität kann im Anschluss Aufschluss darauf geben, ob die Daten nach Durchlauf des Konzepts die gewünschte Qualität erreicht haben. Sollte das Ergebnis nach wie vor nicht zufriedenstellend ausfallen, müssen ggf. strengere Integritätsbedingungen für die Daten definiert werden oder Schritte der Datenbereinigung angepasst werden. In vereinzelten Fällen können händische Maßnahmen zur Nachbearbeitung eingeleitet werden, durch die Werte, die sich automatisch nicht korrigieren ließen, erneut aufgegriffen werden. Ist das Ergebnis jedoch ausreichend, so sollten die Rohdaten nun in entsprechend integrierbarer Form vorliegen und können der Supply-Chain-Datenbank zugefügt werden.

4.2 Validierung des Fehlerkorrektur-Konzepts

Um abschließend die Anwendbarkeit der zusammengestellten Überlegungen zur Fehlerkorrektur von Supply-Chain-Daten zu überprüfen, wird nun anhand einiger Beispieldatensätze (Abschn. 4.2.1) sowohl Fehlerdetektion, als auch Fehlerkorrektur geprobt. Abschließend wird ein Fazit (Abschn. 4.2.2) über den Erfolg der zuvor theoretisch erdachten Vorgehen gezogen und inwiefern sich die Qualität der Daten durch das beschriebene Verfahren steigern ließ.

4.2.1 Datentypen und Datenkategorien der Beispieldatensätze

Die im Folgenden verwendeten Datensätze entstammen der Firma SARMED, einem in Griechenland ansässigen Logistikunternehmen. Damit entstammen die Daten einem Mittelsmann in der Supply-Chain, der als Bindeglied zwischen Produzenten und Endkunden bzw. Händlern steht. Die Daten liegen in der Form von Microsoft EXCEL Daten vor und beinhalten verschieden logistisch relevante Datensätze. Der Fokus der Daten liegt auf Transaktionsdaten und deckt Kunden- und Lieferantendaten ab, aber auch Auftragsdaten und Produktdaten sind in den Datensätzen aufgezeichnet.

Den bereitgestellten Datensätzen sind fünf Tabellen verschiedener Kategorien entnommen worden. Darunter befinden sich die Datensätze „Products“, „Deliveries“, „Nodes“, „Routes“ und „Vehicles“.

1	ProductID	EAN	ItemsPerBox	BoxesPerPallet	BoxVolume	BoxWeight	PrdType	PrdDescr	PrdType2
2	000011	5010327000176	6	50	0	0	ΞΗΡΟ	GLENFIDDICH 12 YO WHISKY 0.7L Χ/6ΦΛ 40%	
3	000015	5011007003005	12	48	0	0	ΞΗΡΟ	JAMESON ΓΥΜΝΟ 0.7L Χ/12ΦΛ 40%	
4	000016	8410024700015	6	95	0	0	ΞΗΡΟ	MALIBU ΓΥΜΝΟ 0.7L Χ/6ΦΛ 21%	
5	000022	8002230000012	6	50	0	0	ΞΗΡΟ	APEROL APERITIVO 1L Χ/6ΦΛ 11%	
6	0000250	5201246002581	24	117	0,01248	8,904	ΞΗΡΟ	Pils Hellas Δίσκος 24 Κουτιών 33 cl 6pack	

Abbildung 4.8: Auszug aus dem Products Datensatz

Der erste Datensatz ist der „Products“ Datensatz (Abb. 4.8), der verfügbare Produkte beschreibt. „*ProductID*“ und „*EAN*“ dienen zur eindeutigen Identifikation von Artikeln. „*ItemsPerBox*“ und „*BoxesPerPallet*“ geben verfügbare Gebindegröße an und „*BoxVolume*“ und „*BoxWeight*“ das spezifische Volumen bzw. Gewicht der Gebinde. Abschließend ist in „*PrdType*“, „*PrdType2*“ und „*PrdDescr*“ Platz für eine Produktbeschreibung vorhanden.

1	ProductID	StartingNodeID	DeliveryNodeID	RouteID	Date	Quantity_items	Quantity_Weight	Quantity (in pallets)	DeliveryTime	OnTime
2	10101012000000	S01	100120501010013/72	82229	03.07.2014	4032	1580,544	0,498113956		1
3	10221087000037	S01	100120501010013/72	82229	03.07.2014	522	630,054	0,822342601		1
4	10255017000025	S01	100120501010013/72	82229	03.07.2014	1008	1250,928	0,741499069		1
5	10261017000025	S01	100120501010013/72	82229	03.07.2014	1440	1627,2	0,374483474		1
6	10802027000146	S01	100120501010013/72	82229	03.07.2014	66	81,906	0,606393628		1

Abbildung 4.9: Auszug aus dem Deliveries Datensatz

Im Datensatz „*Deliveries*“ (Abb. 4.9) werden Angaben zu Lieferungen festgehalten. „*ProductID*“ referenziert dabei ein Produkt, das geliefert wurde. „*StartingNodeID*“ und „*DeliveryNodeID*“ geben IDs zum Start- und Endpunkt der Lieferung wieder. „*RouteID*“ weist der Lieferung einer festen Route und „*Date*“ einem festen Datum zu. „*Quantity_Items*“, „*Quantity_Weight*“ und „*Quantity (in pallets)*“ geben Informationen zur gelieferten Warenmenge. Letztlich gibt „*DeliveryTime*“ die Lieferzeit an und „*OnTime*“ als wahr-oder-falsch-Wert ob die Ware termingerecht geliefert wurde.

	DlsCode	GLN	Address	TK	Coordinates	DeliveryWindow	Description	TypeOfNode
2	00-0904.4		ΑΓ.ΘΩΜΑΣ ΟΙΝΟΗ Τ.Κ. 19012 ΑΤΤΙΚΗ ΕΛΛΑΔΑ	19012	38.1537;23.3437		ΜΑΚΡΟ ΚΑΣ & ΚΑΡΡΥ ΧΑ	Store
3	00-0905.3		ΑΓΙΟΥ ΙΩΑΝΝΟΥ ΘΕΟΛΟΓΟΥ 60 ΑΧΑΡΝΕΣ Τ.Κ. 13672 ΑΤΤΙΚΗ ΕΛΛΑΔΑ	13672	38.103;23.7548		ΜΑΡΙΝΟΠΟΥΛΟΣ ΑΕ	Store
4	00-0956		ΧΛΟΗΣ 92 ΜΕΤΑΜΟΡΦΩΣΗ Τ.Κ. 14452 ΑΤΤΙΚΗ ΕΛΛΑΔΑ	14452	38.0556;23.7584		VINALIA A.E.	Store
5	00-0956.12		ΘΕΣΗ ΛΑΚΚΟ ΚΑΜΑΤΕΡΟ ΑΣΠΡΟΠΥΡΓΟΣ Τ.Κ. 19300 ΑΤΤΙΚΗ ΕΛΛΑΔΑ	19300	38.062;23.5863		LOGISTICS ΑΤΤΙΚΗ ΚΙΝΗΣ	Store
6	00-0956.6		LOGISTIGS ΑΤΤΙΚΗ ΚΙΝΗΣΗ ΘΕΣΗ ΠΑΝΑΓΙΑ ΜΑΝΔΡΑ Τ.Κ. 19600 ΑΤΤΙΚΗ	19600	38.0756;23.5088		VINALIA AE	Store

Abbildung 4.10: Auszug aus dem Nodes Datensatz

In „Nodes“ (Abb. 4.10) sind Stammdaten der Liefer-Knotenpunkte abgelegt. Im Attribut „*Address*“ sind Adressangaben wie Straße und Hausnummer gespeichert. „*TK*“ ist die jeweilige Postleitzahl des Knotenpunktes in Griechenland. In „*Coordinates*“ werden zudem die genauen Geokoordinaten festgehalten, mit denen der genaue globale Standpunkt abgerufen werden kann. „*DeliveryWindow*“ gibt die Möglichkeit eine zeitliche Angabe einzutragen, wann der jeweilige Knotenpunkt beliefert werden kann. Die Attribute „*Description*“ und „*TypeOfNode*“ geben abschließend an, um was für eine Art Knotenpunkt es sich handelt.

	VhrCode	VhcPlateNum	VhrDateTime	CargoVolume	CargoWeight	TotalKlms	ReturnKlms
2	81901	IAE2890	01.04.2015 13:08	0	1600	0	0
3	81902	NXA3069	01.04.2015 13:08	11,55468	410,956	0	0
4	81906	MED FRIGO ΚΑΛΛ 6, 5	01.04.2015 13:08	0	10980,68	0	0
5	81907	NXA2641	01.04.2015 13:08	2,0328	4983,33	0	0
6	81909	NXA3590	01.04.2015 13:08	0	31056,85	0	0

Abbildung 4.11: Auszug aus dem Routes Datensatz

Der vierte tabellarische Datensatz „*Routes*“ (Abb. 4.11) beschreibt Angaben zu den Lieferrouten des Unternehmens. Im ersten Attribut „*VhrCode*“ wird der Route ein eindeutiger Code zugewiesen. „*VhcPlateNum*“ beschreibt das Kennzeichen des zur Lieferung eingesetzten Transportfahrzeuges. Die zeitliche Angabe, wann die Route gefahren wurde, wird anschließend in „*VhrDateTime*“ angegeben. „*CargoVolume*“ und „*CargoWeight*“ zeigen spezifische Angaben zu Volumen und Gewicht der transportierten Ware. Abschließend können gefahrene Kilometer der Hin- und Rückstrecke in „*TotalKlms*“ und „*ReturnKlms*“ eingetragen werden.

	VehicleID	Contracted	TypeOfVehicle	EngineTechnology	EngineGas	VehicleGrossWeight	VehicleNetWeight	MaxCargoVolume
2	NXA5926	1	7 ΠΑΛΕΤΕΣ	Euro 2	Diesel	0	2850.00	7 ΠΑΛΕΤΕΣ
3	NXA5494	1	15 ΠΑΛΕΤΕΣ	Euro 3		0	2770.00	15 ΠΑΛΕΤΕΣ
4	NXA1716	1	8 ΠΑΛΕΤΕΣ	Euro 1	Diesel	0	1300.00	8 ΠΑΛΕΤΕΣ
5	NXA5132	1	9 ΠΑΛΕΤΕΣ	Euro 1		0	2500.00	9 ΠΑΛΕΤΕΣ
6	NXA2617	1	12 ΠΑΛΕΤΕΣ	Euro 3	Diesel	0	2940.00	12 ΠΑΛΕΤΕΣ

Abbildung 4.12: Auszug aus dem Vehicles Datensatz

Der letzte Datensatz (Abb. 4.12) ist „Vehicle“ und führt Daten zu den Lieferfahrzeugen. Im ersten Attribut, das „VehicleID“ heißt, werden eindeutige Identifikationsnummern zu den Fahrzeugen festgehalten. Das Feld „Contracted“ gibt als wahr-oder-falsch-Wert an, ob es sich bei dem Fahrzeug um Firmeneigentum handelt. „TypeOfVehicle“ enthält Informationen zum Fahrzeugtypen. „EngineTechnology“ beschreibt die Schadstoffklasse des Fahrzeugs und „EngineGas“ den Treibstofftypen. Im Attribut „VehicleGrossWeight“ wird das zulässige Gesamtgewicht des Fahrzeuges angegeben. „VehicleNetWeight“ gibt das Leergewicht des Fahrzeugs an. Abschließend wird im Attribut „MaxCargoVolume“ die beförderbare Menge an Waren in Volumen angegeben.

Zunächst können nun die einzelnen Attribute der Datenquellen anhand ihrer Funktion, nach dem Vorgehen von Ziegler (2015), zwecks Kategorisierungsansatz, sortiert werden.

Funktion	Bezeichnung
Gebindegröße	ItemsPerBox; BoxesPerPallet
Gewicht	BoxWeight
Lieferadresse	Adress; TK; Coordinates; Description; TypeOfNode
Maße	BoxVolume
Max. Kapazität	VehicleGrossWeight; VehicleNetWeight; MaxCargoVolume
Menge	Quantity_Items; Quantity_Weight; Quantity (In Pallets); Cargo_Volume; Cargo_weight
Produktkennung	EAN; PrdType; PrdType2; PrdDescr
Ressourcenkennung	TotalKlms; ReturnKlms; Contracted; TypeOfVehicle; EngineTechnology; EngineGas
Schlüssel	ProductID; DeliveryNodeID; StartingNodeID; RouteID; VhcPlateNum; VhrCode; VehicleID
Zeitstempel	Date; DeliveryTime; OnTime; VhrDateTime
Zeitstempel/ Liefertermin (Wunsch)	DiliveryWindow

Tabelle 4.4: Funktionszuweisung der Attribute nach Zie15

Fast jedem Attribut der Datensätze konnte eine Funktion zugewiesen werden. Ausgenommen sind die Attribute „DlsCode“ und „GLN“ aus dem „Nodes“ Datensatz. Dieser Umstand wird im folgenden Abschnitt erneut aufgegriffen und erläutert. Im Anschluss werden die Datentypen nach dem Vorgehen von Ziegler (2015) kategorisiert, um die spätere Fehlerkorrektur zu vereinfachen.

Kategorie	Funktion
Referenzdaten *(fix)*	
Unternehmensstrukturdaten *(fix)*	
Transaktionsstrukturdaten *(fix)*	• Lieferadresse • Produktkennung • Ressourcenkennung • Max. Kapazität
Bestandsdaten *(variabel)*	• Menge
Transaktionsaktivitätsdaten	• Menge • Maße • Gewicht • Gebindegröße
Transaktionskontrolldaten	• Zeitstempel • Zeitstempel/ Liefertermin (Wunsch)
Metadaten	• Produktkennung
Hierarchische Daten	• Schlüssel
Unstrukturierte Daten	• Lieferadresse • Produktkennung

Tabelle 4.5: Kategorisierung der Beispieldaten nach Zie15

Auch hier lässt das Kategorisierungsmodell von Ziegler (2015) eine eindeutige Zuordnung aller Funktionen in Kategorien zu. Damit sind alle Datentypen der vorliegenden Datensätze erfolgreich kategorisiert worden. Die Rohdaten der zu integrierenden Datensätze liegen nun in geordneter Darstellung vor und können bei auftretenden Fehlern anhand ihrer Kategorie behandelt werden bzw. lassen die vorliegenden Kategorien direkt einen Schluss auf die Korrigierbarkeit der einzelnen Datenfehler zu.

4.2.2 Fehlerkorrektur eines Beispieldatensatzes

Damit sind die grundlegenden Formen der Datensätze bekannt. Zunächst wird geschildert, in welcher Form und in welchem Zustand die Daten in den entsprechenden Tabellen der Excel Dokumente vorliegen. Im Anschluss folgt eine Überprüfung, welcher Aufwand zur Datenharmonisierung, zwecks Integration in bestehende Systeme betrieben werden muss und wie schrittweise der Ablauf dargestellt werden kann. Anschließend wird auf den qualitativen Sachverhalt der Daten in Bezug auf Vollständigkeit und Fehlerbehaftung eingegangen. Als abschließender Schritt werden die zuvor erarbeiteten Maßnahmen exemplarisch auf gefundene Fehler angewandt und versucht, eine Senkung des Fehlermaßes und eine Steigerung der Benutzbarkeit der Daten zu bewerkstelligen.

Um nun die Daten einheitlich nutzen und in ein zentrales System, wie ein Data-Warehouse, integrieren zu können, muss zunächst geklärt werden, welche Schritte notwendig sind, um eine homogene Form zu erhalten. Das heißt, eine Überprüfung der vorliegenden Datenschemata ist notwendig, um mittels Schema-Matching-Verfahren eine einheitlich nutzbare Form zwischen Rohdaten und bereits bestehenden Daten in der Supply-Chain, aber auch den unterschiedlichen Rohdatensätzen selber zu erzeugen.

Die Daten, die einem griechischen Unternehmen entstammen, weisen in diesem Fall nicht nur eine andere Sprache der textuellen Dateneinträge auf, sondern auch eine Zeichenkodierung basierend auf griechischem Zeichensatz.

Damit wird bereits zu Beginn klar, dass Heterogenität zwischen den Datensätzen und dem Zielsystem besteht. Damit die Einträge verarbeitet und ggf. übersetzt werden können, muss zunächst das griechische Alphabet ins lateinische Alphabet überführt werden.

Eine einfache Parsing-Tabelle hilft dabei bei der Transformation des jeweiligen Zeichens. Die von Excel unterstützte Nutzung von Visual Basic Modulen ermöglicht eine dokumentenweite Überführung der Zeichen.

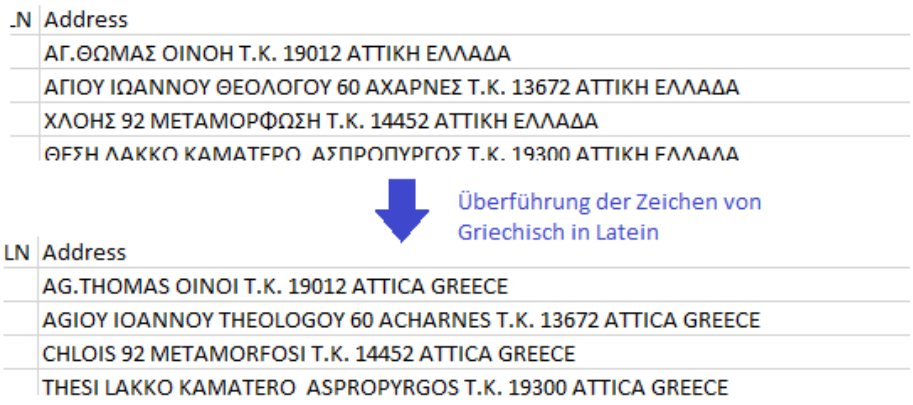


Abbildung 4.13: Textzeichenüberführung & Übersetzung am Beispiel von Datensatz „Nodes“

Wichtig bei der Konvertierung ist, dass Visual Basic basierend auf der Unicode-Codierung, keine direkte Möglichkeit bereitstellt, griechische Zeichen darzustellen. Diese müssen mit Hilfe der ChrW()-Funkton über ihre Unicodenummer dargestellt werden. Eine Erweiterung der Tabelle 4.1 zeigt die jeweilige Codierung aller Zeichen:

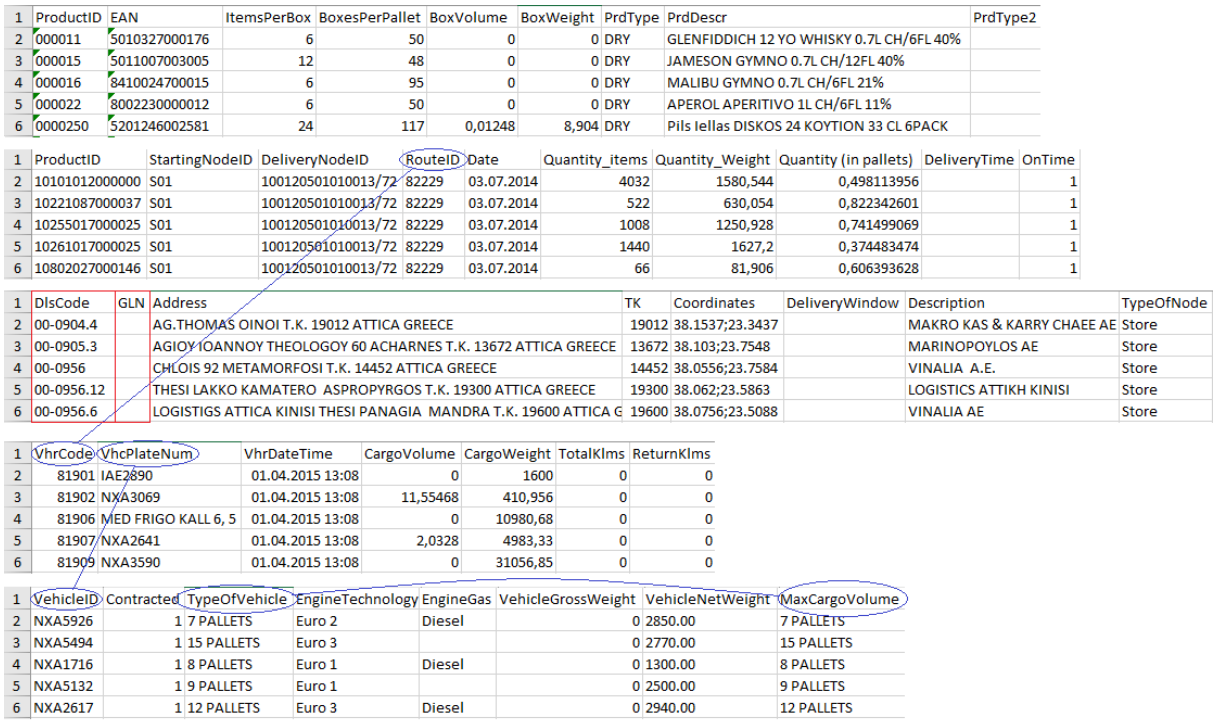
Griechischer Zeichensatz	Unicodenummern	Lateinischer Zeichensatz
Α, α	913, 945	A, a
Β, β, ϐ	914, 946, 976	B, b
Γ, γ	915, 947	G, g
Δ, δ	916, 948	D, d
Ε, ε, ϵ	917, 949, 1013	E, e
Ζ, ζ	918, 950	Z, z
Η, η	919, 951	I, i
Θ, θ, ϑ	920, 952, 977	TH, th
Ι, ι	921, 953	I, i
Κ, κ, ϰ	922, 954, 990	K, k
Λ, λ	923, 955	L, l
Μ, μ	924, 956	M, m
Ν, ν	925, 957	N, n
Ξ, ξ	926, 958	X, x
Ο, ο	927, 959	O, o
Π, π, ϖ	928, 960, 982	P, p
Ρ, ρ, ϱ	929, 961	R, r
Σ, σ, ς, ϲ	931, 962, 962, 1010	S, s
Τ, τ	932, 964	T, t
Υ, υ	933, 965	Y, y
Φ, ϕ, φ	934, 981, 966	F, f
Χ, χ	935, 967	CH, ch
Ψ, ψ	936, 968	PS, ps
Ω, ω	937, 969	O, o

Tabelle 4.6: Überführungstabelle Zeichensatz „Griechisch-Lateinisch“

Neben Heterogenität durch Nutzung eines anderen Zeichensatzes fallen weitere Fälle von semantischer Heterogenität beim Vergleich der verschiedenen Datensätze miteinander auf (vgl. Abschn. 3.1.2). So sind sowohl Fälle von Synonymen, als auch von Homonymen vorhanden. Zudem sind weitere Begriffsungenauigkeiten wie undefinierte Begriffe und unklare Abkürzungen in den Datensätzen vorzufinden. Im „*Deliveries*“ wird das Attribut, dass der Route eine eindeutige ID zuweist, mit „RouteID“ bezeichnet. Im Datensatz „*Routes*“ wird die ID der Route im Attribut „*VhrCode*“ gespeichert. Bei einer Zusammenführung der beiden Datensätze muss zur Vergleichbarkeit anhand der Routen-ID eine einheitliche Bezeichnung gewählt werden. Da „*RouteID*“ eine eindeutigere Bezeichnung ist, wird sie für beide Datensätze gewählt. Die gleiche Problematik herrscht zwischen den Datensätzen „*Vehicles*“ und „*Routes*“.

Die eindeutige Zuordnung der Fahrzeuge erfolgt über das Kennzeichen, das im Datensatz „*Vehicles*“ unter „*VehicleID*“ geführt wird, in „Routes“ jedoch unter dem Attribut „*VhcPlateNum*“. Auch hier wird mit zwei Bezeichnern die gleiche Eigenschaft beschrieben. Als einheitlicher Bezeichner wird „*VehicleID*“ gewählt.

Im Zuge der Überprüfung auf semantische Heterogenität fallen mehrere unklare Elemente in den Datensätzen auf. In diesem Fall handelt es sich um Attribute, denen mangels Kontext keine Funktion bzw. Bedeutung zugeschrieben werden kann. Im Datensatz „Nodes“ befinden sich die Attribute „DlsCode“ und „GLN“, die ohne weitere Informationen vom Ersteller der Datei unbekannte Bedeutung behalten. Da diese Informationen in dem Zustand keinen ersichtlichen Nutzen für die restlichen bestehenden Daten haben, werden diese aus dem Schema entfernt.



ProductID	EAN	ItemsPerBox	BoxesPerPallet	BoxVolume	BoxWeight	PrdType	PrdDescr	PrdType2
000011	5010327000176	6	50	0	0	DRY	GLENFIDDICH 12 YO WHISKY 0.7L CH/6FL 40%	
000015	5011007003005	12	48	0	0	DRY	JAMESON GYMNO 0.7L CH/12FL 40%	
000016	8410024700015	6	95	0	0	DRY	MALIBU GYMNO 0.7L CH/6FL 21%	
000022	8002230000012	6	50	0	0	DRY	APEROL APERITIVO 1L CH/6FL 11%	
0000250	5201246002581	24	117	0,01248	8,904	DRY	Pils Iellas DISKOS 24 KOYTION 33 CL 6PACK	

ProductID	StartingNodeID	DeliveryNodeID	RouteID	Date	Quantity_items	Quantity_Weight	Quantity (in pallets)	DeliveryTime	OnTime
10101012000000	S01	1001205010100l3/72	82229	03.07.2014	4032	1580,544	0,498113956		1
10221087000037	S01	1001205010100l3/72	82229	03.07.2014	522	630,054	0,822342601		1
10255017000025	S01	1001205010100l3/72	82229	03.07.2014	1008	1250,928	0,741499069		1
10261017000025	S01	1001205010100l3/72	82229	03.07.2014	1440	1627,2	0,374483474		1
10802027000146	S01	1001205010100l3/72	82229	03.07.2014	66	81,906	0,606393628		1

DlsCode	GLN	Address	TK	Coordinates	DeliveryWindow	Description	TypeOfNode
00-0904.4		AG.THOMAS OINOI T.K. 19012 ATTICA GREECE	19012	38.1537;23.3437		MAKRO KAS & KARRY CHAEE AE	Store
00-0905.3		AGIOY IOANNOY THEOLOGOY 60 ACHARNES T.K. 13672 ATTICA GREECE	13672	38.103;23.7548		MARINOPOYLOS AE	Store
00-0956		CHLOIS 92 METAMORFOSI T.K. 14452 ATTICA GREECE	14452	38.0556;23.7584		VINALIA A.E.	Store
00-0956.12		THESI LAKKO KAMATERO ASPROPYRGOS T.K. 19300 ATTICA GREECE	19300	38.062;23.5863		LOGISTICS ATTIKH KINISI	Store
00-0956.6		LOGISTIGS ATTICA KINISI THESI PANAGIA MANDRA T.K. 19600 ATTICA G	19600	38.0756;23.5088		VINALIA AE	Store

VhrCode	VhcPlateNum	VhrDateTime	CargoVolume	CargoWeight	TotalKlms	ReturnKlms
81901	IAE2890	01.04.2015 13:08	0	1600	0	0
81902	NXA3069	01.04.2015 13:08	11,55468	410,956	0	0
81906	MED FRIGO KALL 6, 5	01.04.2015 13:08	0	10980,68	0	0
81907	NXA2641	01.04.2015 13:08	2,0328	4983,33	0	0
81909	NXA3590	01.04.2015 13:08	0	31056,85	0	0

VehicleID	Contracted	TypeOfVehicle	EngineTechnology	EngineGas	VehicleGrossWeight	VehicleNetWeight	MaxCargoVolume
NXA5926	1	7 PALLETS	Euro 2	Diesel	0	2850.00	7 PALLETS
NXA5494	1	15 PALLETS	Euro 3		0	2770.00	15 PALLETS
NXA1716	1	8 PALLETS	Euro 1	Diesel	0	1300.00	8 PALLETS
NXA5132	1	9 PALLETS	Euro 1		0	2500.00	9 PALLETS
NXA2617	1	12 PALLETS	Euro 3	Diesel	0	2940.00	12 PALLETS

Abbildung 4.14: Heterogenität der Rohdatensätze (Synonyme in blau; Unklare Elemente in rot)

Das weitere Vorgehen in der Datenbearbeitung wird, wie zuvor in Abschn. 4.1.1 beschrieben, fortgeführt. Zunächst wird die Datenheterogenität überwunden, um Zugang zu den einzelnen Daten zu bekommen. Dieser Schritt wurde im Vorangegangenen beschrieben. Im Anschluss müssen die formal angepassten Daten noch von Datenfehlern bereinigt werden.

Dazu wird zunächst versucht, Datenfehler innerhalb der einzelnen Datenquellen zu lokalisieren und diese im Anschluss zu beseitigen. Abschließend werden Datenfehler zwischen den zu integrierenden Datenquellen gesucht und diese mithilfe entsprechender Maßnahmen beseitigt.

Bei Einzeldatenfehlern kommt es, wie schon in Abschn. 3.2.3 beschrieben, auf die Fehler innerhalb der einzelnen Datenquellen an. Korrespondenzen zwischen den Datensätzen bleiben an diesem Punkt unbeachtet. Datenfehler, die in diesem Zuge gesucht werden und zu korrigieren versucht werden, sind:

Auf Schemaebene:	Auf Datenebene
- Unzulässige Werte - Attributabhängigkeiten verletzt - Eindeutigkeit verletzt - Referenzielle Integrität verletzt	- Fehlende Werte - Schreibfehler - Falsche Werte - Falsche Referenz - Kryptische Werte - Eingebettete Werte - Falsche Zuordnungen - Widersprüchliche Werte - Transpositionen - Duplikate - Datenkonflikte

Tabelle 4.7: Datenfehler im Überblick

Im Zuge der strukturellen Analyse werden strukturierte Datenwerte mittel Musteranalyse auf Fehler zu überprüfen und Null-Werte in der gesamten Datenmenge zu identifizieren.

Ein häufig auftretender Fehler, der meist durch fehlerhaftes Vorgehen beim Erstellen der Datensätze herrührt, sind fehlende Werte oder das Befüllen von Datenattributen mit Dummy-Werten (vgl. Abschn. 3.1.3). Durch die Suche nach „Null-Werten“, also ungefüllten Feldern im Datensatz können fehlende Werte schnell lokalisiert werden. Für das Ausmachen von Dummy-Werten muss ein Kontext zum Feld bekannt sein, um eine automatische Unterscheidung zwischen Real- und Dummy-Wert zu ermöglichen. Abbildung 4.15 zeigt einige Null-Werte innerhalb des „Vehicles“ Datensatzes. Diese textuellen Angaben fehlen in den entsprechenden Zeilen vollständig und werden direkt bei einer Suche nach leeren Attributen mithilfe der =ISTLEER() Anweisung gefunden.

	A	B	C	D	E	F	G	H
1	VehicleID	Contracted	TypeOfVehicle	EngineTechnology	EngineGas	VehicleGrossWeight	VehicleNetWeight	MaxCargoVolume
2	NXA5926	1	7 PALLETS	Euro 2	Diesel	0	2850.00	7 PALLETS
3	NXA5494	1	15 PALLETS	Euro 3		0	2770.00	15 PALLETS
4	NXA1716	1	8 PALLETS	Euro 1	Diesel	0	1300.00	8 PALLETS
5	NXA5132	1	9 PALLETS	Euro 1		0	2500.00	9 PALLETS
6	NXA2617	1	12 PALLETS	Euro 3	Diesel	0	2940.00	12 PALLETS
7	NXA3172	1	12 PALLETS	Euro 1	Diesel	0	5000.00	12 PALLETS
8	NXA4999	1	8 PALLETS	Euro 3		0	3600.00	8 PALLETS
9	NXA1025	1	10 PALLETS	Euro 3	Diesel	0	3440.00	10 PALLETS
10	NXA6857	1	12 PALLETS	Euro 4		0	3800.00	12 PALLETS
11	NXA1171	1	15 PALLETS	Euro 4		0	2550.00	15 PALLETS
12	NXA5417	1	8 PALLETS	Euro 2		0	3800.00	8 PALLETS
13	NXA4761	1	15 PALLETS	Euro 5		0	3300.00	15 PALLETS
14	EKB4382	1	0 PALLETS	Euro 6		0	1520.00	0 PALLETS
15	NXA5178	1	12 PALLETS	Euro 3	Diesel	0	1010.00	12 PALLETS
16	NXA1471	1	8 PALLETS	Euro 2	Diesel	0	2010.00	8 PALLETS
17	NXA3140	1	6 PALLETS	Euro 4	Diesel	0	1500.00	6 PALLETS
18	NXA4249	1	10 PALLETS	Euro 1		0	1600.00	10 PALLETS
19	NXA2417	1	15 PALLETS	Euro 1	Diesel	0	920.00	15 PALLETS
20	NXA5858	1	12 PALLETS			0	0.00	12 PALLETS
21	NXA2205	1	4 PALLETS	Euro 2	Diesel	0	858.00	4 PALLETS

Abbildung 4.15: Nullwerte in Vehicles Datensatz

Die Musteranalyse kann besonders bei Strukturdaten angewandt werden. Darunter fallen in diesen Rohdaten „EAN“ aus „Products“, „TK“ aus „Nodes“, „VhcPlateNum“ aus „Routes“, „VehicleID“ aus „Vehicles“ und als Zeitstempel „Date“ aus „Deliveries“ und „VhrDateTime“ aus „Routes“. Für das Attribut „EAN“ wurde in diesem Fall EAN-13 eingetragen. Da EAN-13 eine feste genormte Form hat, lässt sich die Mustererkennung gut auf dieses Attribut anwenden.

In EAN-13 sind Ziffern von 0-9 erlaubt und die Codierung weist genau 13 Ziffern auf. Zudem lässt sich leicht bestimmen, ob die EAN Tippfehler enthält. EAN-13 ist wie folgt aufgebaut (siehe Abb.4.16)

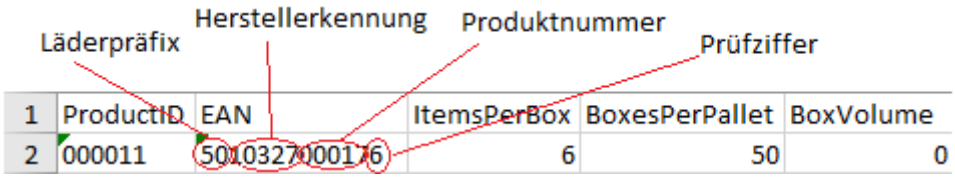


Abbildung 4.16: Aufbau der EAN-13 Nummer (eigene Darstellung)

Mittels der Prüfziffer lässt sich berechnen, ob die dargestellte EAN korrekt ist. Dies geschieht per Modulo-10 Berechnung:

EAN-Nummer	501032700017
Prüfziffer	6
Ziffern	5 0 1 0 3 2 7 0 0 0 1 7
Multiplikator	1 3 1 3 1 3 1 3 1 3 1 3
Ergebnis	5 0 1 0 3 6 7 0 0 0 1 21 = (Quersumme) 44
Prüfsumme	Differenz des Ergebnisses zum nächstgrößeren Vielfachen von 10 = 6

Tabelle 4.8: Berechnung der EAN-13 Nummer (eigene Darstellung nach [ABC])

Für diesen Fall ist das Ergebnis übereinstimmend. Es kann davon ausgegangen werden, dass es sich tatsächlich um eine gültige EAN-Nummer handelt.

Für „VehicleID“ bzw „VhcPlateNum“ lassen sich ähnliche Überprüfungen der Struktur anwenden. Die Darstellung der Kennzeichen Griechenlands erfolgt, ähnlich wie in Deutschland, nach festem Muster. Griechische Kennzeichen bestehen immer aus drei Großbuchstaben, gefolgt von 4 Ziffern (1000-9999).

Als Buchstaben werden nur Buchstaben verwenden, die sowohl im lateinischen, als auch griechischen Alphabet vorkommen (A, B , E, H, I, K, M ,N, O, P, T, X, Y, Z) [ITE]. Anhand dieser strukturellen Vorgaben können die Einträge direkt während der strukturellen Analyse überprüft werden.

Fehlende Werte, die mit Dummy-Werten gefüllt wurden, sind deutlich schwieriger auszumachen. Bei numerischen Werten können mit Hilfe von Gültigkeitsbereichen Dummy-Werte lokalisiert werden. Aus diesem Grund wird versucht, mithilfe von Bedingungen zu den Datenwerten der Datensätze Dummy-Werte und Ausreißer zu lokalisieren. Der einfachste Fall ist bei numerischen Attributen der „0-Wert“

28	0002155		24	108	0,01248	8,904	DRY	PREMIUM PILSENER "Z" KOYTI 33 CL (3+1 DORO)6CH4PACK
29	0002156		24	108	0,01248	8,904	DRY	PREMIUM PILSENER "Z" KOYTI 33 CL (4+2 DORO) 6PACK
30	0002191		0	1	0,05184	39,5	DRY	PREMIUM PILSENER "Z" BARELI 30LT
31	0002192		0	1	0,040176	29,1	DRY	PREMIUM PILSENER "Z" BARELI 20LT
32	000227	5010327000176	6	50	0	0	DRY	GLENFIDDICH 12 YO WHISKY KYLINDROS 0.7L CH/6FL 40%
33	000300	5902573004568	12	50	0	0	DRY	PR STOCK PRESTIGE VODKA 0.7L CH/12FL 40%
34	000306	3049197210202	12	25	0	0	DRY	PR COURVOISIER V.S.O.R 0.7L CH/12FL 40%
35	000307	3049197210639	6	25	0	0	DRY	PR COURVOISIER V.S.O.R 0.7L KOYTI+2GLASS CH/6FL 40%
36	0003200		0	1	0,015484	19,2	DRY	CO2 PERIECHOMENO 6 KG
37	0003202		0	1	0,031212	25,3	DRY	CO2 PERIECHOMENO 10 KG AL.Luxfer
38	000344	721733000999	3	10	0	0	DRY	PATRON GRAN BURDEOS 0.7L CH/3FL 40%
39	000372	721733000715	6	50	0	0	DRY	PATRON ANEJO TEQUILA 0.7L CH/6FL 40%
40	0004050	5201246002215	20	50	0,03264675	20,6	DRY	Pils Iellas KIBOTIO 20 EF 50 CL

Abbildung 4.17: Dummy-Werte und Fehlende Werte im Auszug aus „Products“

Im Datensatz „Products“, in dem Angaben zu den verfügbaren Produkten vorliegen, sind mithilfe von Gültigkeitsbereichen für Werte der numerischen Produktdaten Dummy-Werte zu identifizieren. So werden „0-Werte“ in Größen-, Gewichts- und Gebindeangaben der Einzelprodukte markiert, sobald die Bedingung gesetzt wurde, dass diese einen positiven Wert größer 0 haben müssen.

Im Zuge der Normalisierung von Datenformaten wird nun versucht, eine einheitliche und stimmige Darstellung der Daten zu erstellen. Unstimmige Formate, die im Zuge der strukturellen Analyse ermittelt wurden, werden an dieser Stelle korrigiert. Aber auch unstrukturierte textuelle Attribute, die zur Weiterverarbeitung und Korrektur in strukturierter Form benötigt werden. Der Datensatz „Nodes“ enthält ein solches Attribut. In diesem Fall ist im Attribut „Address“ der unformatierte Adresssatz der einzelnen Einträge gespeichert.

1900		OYLOF PALME 15 ZOGRAFOS ATHENS 15771 2107714029 PROIOS N. & SIA O.E.	15771	37.9769;23.7616
1901		GRIGORI LAMRAKI 34 TAYROS ATHENS 17778 2103476227 KARAGILANI GEORGIA	17778	38.0071;23.7276
1902		ARISTOTELOYS 187 ACHARNES ATHENS 13675 2102484457 CHATZEIPIDIS A. THOMAS	13675	38.0898;23.7384
1903		KATECHAKI 39 AMPELOKIPOI ATHENS 11525 2106913564 KOLAXIDIS TH. KAI KOLAXIDIS I. O.E.	11525	37.9938;23.7751
1904		ARAPAKI 18-20 KALLITHEA ATHENS 17676 2109571956 SOYLIOY AGGELIKI KAI SIA EE	17676	37.9596;23.7058
1905		L.LAYRIOY 45 PALLINI PALLINI 15351 2106040545 ANDRIKOPOYLOY AIKATERINI & THEODORA O.E.	15351	37.9844;23.8513
1906		LYKOYRGOY 18 & SOKRATOYS PL. KOTZIA ATHENS 10552 2105235722 LOGOTHETIS A. & SIA O.E.	10552	37.9829;23.7264
1907		ELPIDOS 10 NEA IONIA ATHENS 14231 2102799174 KANDRALIDIS MINAS	14231	38.0453;23.7558
1908		AGIOY ELEYTHERIOY 139 KAMINIA PEIRAIAS 18541 2104835961 SCHOINA G. YIOI OE	18541	37.9572;23.663
1909		MARKOY MPOTSARI 22 AIGALEO ATHENS 12241 2105906257 AFOI ASIMAKOPOYLOI O.E.	12241	37.991;23.6797
1910		PROPONTIDOS 14 NEA PALATIA NEA PALATIA 19015 2295032250 DIOMATARIS IL. & SIA O.E.	19015	38.3217;23.7985

Abbildung 4.18: Adresszeile im „Nodes“ Datensatz vor der Normalisierung

In diesem Auszug der Adressdaten aus dem Datensatz „Nodes“ wird ersichtlich, dass die Adressdaten mit Konvertierungsfehlern durchsetzt sind. Unter anderem werden unregelmäßige Worttrennungen, fehlende Trennzeichen oder doppelte Informationen verwendet. Zudem sind teilweise Informationen anderer Attribute, wie der Postleitzahl aus „TK“ vorhanden und Informationen, die nicht zu den Adressdaten gehören, wie Telefonnummern. Durch eine Konvertierung mittels Visual Basic Moduls kann ein einheitliches und zu Weiterverarbeitung taugliches Bild geschaffen werden (siehe Anhang I).

1900		OYLOF PALME 15 ZOGRAFOS ATHENS PROIOS N. & SIA O.E.	15771	37.9769;23.7616
1901		GRIGORI LAMRAKI 34 TAYROS ATHENS KARAGILANI GEORGIA	17778	38.0071;23.7276
1902		ARISTOTELOYS 187 ACHARNES ATHENS CHATZEIPIDIS A. THOMAS	13675	38.0898;23.7384
1903		KATECHAKI 39 AMPELOKIPOI ATHENS KOLAXIDIS TH. KAI I. O.E.	11525	37.9938;23.7751
1904		ARAPAKI 18 - 20 KALLITHEA ATHENS SOYLIOY AGGELIKI KAI SIA EE	17676	37.9596;23.7058
1905		L.LAYRIOY 45 PALLINI ANDRIKOPOYLOY AIKATERINI & THEODORA O.E.	15351	37.9844;23.8513
1906		LYKOYRGOY 18 & SOKRATOYS PL. KOTZIA ATHENS LOGOTHETIS A. SIA O.E.	10552	37.9829;23.7264
1907		ELPIDOS 10 NEA IONIA ATHENS KANDRALIDIS MINAS	14231	38.0453;23.7558
1908		AGIOY ELEYTHERIOY 139 KAMINIA PEIRAIAS SCHOINA G. YIOI OE	18541	37.9572;23.663
1909		MARKOY MPOTSARI 22 AIGALEO ATHENS AFOI ASIMAKOPOYLOI O.E.	12241	37.991;23.6797
1910		PROPONTIDOS 14 NEA PALATIA DIOMATARIS IL. & SIA O.E.	19015	38.3217;23.7985

Abbildung 4.19: Adresszeile aus Abb.4.20 nach Anwendung des Normalisierungsmoduls

Die Konvertierung passt im Anschluss Dateneinheiten numerischer Datenwerte an das gewünschte Schema an. Dazu sind zunächst grundlegende Kontextinformationen zu den Informationen notwendig. In den Datensätzen fällt auf, dass numerischen Werte meist keine Angaben zur verwendeten Einheit enthalten. Damit Einheiten entsprechend konvertiert werden können, muss Rückschluss auf die Einheit gezogen werden.

	ProductID	EAN	ItemsPerBox	BoxesPerPallet	BoxVolume	BoxWeight	PrdType	PrdDescr
2	000011	5010327000176	6	50	0	0	DRY	GLENFIDDICH 12 YO WHISKY 0.7L CH/6FL 40%
3	000015	5011007003005	12	48	0	0	DRY	JAMESON GYMNO 0.7L CH/12FL 40%
4	000016	8410024700015	6	95	0	0	DRY	MALIBU GYMNO 0.7L CH/6FL 21%
5	000022	8002230000012	6	50	0	0	DRY	APEROL APERITIVO 1L CH/6FL 11%
6	0000250	5201246002581	24	117	0,01248	8,904	DRY	Pils Iellas DISKOS 24 KOYTION 33 CL 6PACK
7	0000253	5201246002598	24	81	0,01768	12,792	DRY	Pils Iellas DISKOS 24 KOYTION 50 CL 4PACK

Abbildung 4.20: Fehlende Einheiten in Datensatz Products

In Abbildung 4.20 sind für die Attribute „BoxVolume“ und „BoxWeight“ keine Einheiten angegeben. Daher muss aus dem Kontext der Daten erschlossen werden, um welche Einheit es sich bei den Eingangsdaten handelt, um diese entsprechend dem gewünschten Zielformat zu konvertieren. Aus der Produktbeschreibung („PrdDescr“) kann ermittelt werden, in welcher Einheit die Angaben in etwa angeben wurden. Für „BoxWeight“ wären mögliche logische Angaben des Gewichts in Pfund, Kilogramm oder Tonnen. Aus Zeile 6 z.B. geht hervor, dass es sich um 33CL Getränkedosen handelt, die zu 24 Stück („ItemsPerBox“) in einer Box sind. Daher kann von einem ungefähren Nettogewicht von 24x0,33kg = 7,92kg ausgegangen werden. Inklusive Verpackungsmaterialien kann daher davon ausgegangen werden, dass es sich bei 8,904 um eine Angabe in Kilogramm handelt und somit das Attribut „BoxWeight“ komplett in Kilogramm angegeben ist. „BoxVolume“ lässt sich ähnlich ableiten und kommt zu der Annahme, dass es sich um Kubikmeter handelt, ein durchaus gängiges Format in dem Volumenangaben von Transportfahrzeugen angegeben werden. Mit Hilfe dieser Angaben lassen sich die Datenformate, wenn nötig, umrechnen und die Anzahl an Nachkommastellen anpassen.

1	ProductID	EAN	ItemsPerBox	BoxesPerPallet	BoxVolume (m³)	BoxWeight (kg)	PrdType	PrdDescr
2	000011	5010327000176	6	50	0,0000	0,00	DRY	GLENFIDDICH 12 YO WHISKY 0.7L CH/6FL 40%
3	000015	5011007003005	12	48	0,0000	0,00	DRY	JAMESON GYMNO 0.7L CH/12FL 40%
4	000016	8410024700015	6	95	0,0000	0,00	DRY	MALIBU GYMNO 0.7L CH/6FL 21%
5	000022	8002230000012	6	50	0,0000	0,00	DRY	APEROL APERITIVO 1L CH/6FL 11%
6	0000250	5201246002581	24	117	0,0125	8,90	DRY	Pils Iellas DISKOS 24 KOYTION 33 CL 6PACK
7	0000253	5201246002598	24	81	0,0177	12,79	DRY	Pils Iellas DISKOS 24 KOYTION 50 CL 4PACK
8	0000257		24	117	0,0125	8,90	DRY	Pils Iellas DISKOS 24 KOYTION 33 CL 6PACK (4+2 DORO)
9	0000258		24	81	0,0177	12,79	DRY	PILS HELLAS DISKOS 4K CH 6PACK 50CL (4+2 DORO)
10	0000260		24	54	0,0208	14,07	DRY	PILS HELLAS NRB 330ML CH/24FL 6PACK
11	000040		6	60	0,0000	0,00	DRY	LIMONCINO TRADIZIONALE MARCATI 2L CH/6FL 28%
12	000043	5099873045367	12	50	0,0000	0,00	DRY	JACK DANIELS 1L CH/12FL 40%
13	0000557		24	108	0,0125	8,90	DRY	Berlin DISKOS 24 KOYTION 33 CL 6PACK (4+2 DORO)
14	0000558		24	72	0,0177	12,79	DRY	Berlin DISKOS 24 KOYTION 50 CL 6PACK (4+2 DORO)

Abbildung 4.21: Ergänzung der Dateneinheit und Konvertierung der Darstellungsweise in Datensatz Products

Im Anschluss an die Konvertierung wird sich der zuvor als fehlend markierten Einträge in den Datensätzen angenommen. Wie bereit angemerkt, muss die Korrektur an das Vorkommen des Fehlers angepasst werden. Ein anschauliches Beispiel für die Korrektur von fehlenden Werten, an dem sich das vorgestellte Konzept überprüfen lässt, bietet hier der „Nodes“ Datensatz, indem vermehrt fehlende Einträge in den geografischen Angaben wie Adresse oder Geokoordinaten vorhanden sind. Dieser Fehler wird mithilfe des in Abschnitt 4.1.1 erwähnten Geocodings behoben. Mithilfe eines Visual Basic Moduls werden die Adressangaben aus dem Excel-Formular gelesen und mit der Datenbank des API-Anbieters abgeglichen. Bei Übereinstimmung wird ein Breitengrad und ein Längengrad zurückgegeben. Hier wird bereits die Notwendigkeit des zu Beginn getätigten Überführens der Zeichensätze klar. Durch die Überführung der Zeichen in Unicode-darstellbare Zeichen, kann der Eintrag in der Adresszeile durch das Visual Basic Modul verarbeitet werden. Genutzter VB-Code wird in Abbildung 4.22 gezeigt.

```
Function GetGeo(Address As String) As String

    Dim Request             As New XMLHTTP30
    Dim Results             As New DOMDocument30
    Dim Stat, Lat, Lon      As IXMLDOMNode

    On Error GoTo fehler

    Request.Open "GET", "https://maps.googleapis.com/maps/api/geocode/xml?" _
    & "&address=" & Address & "&sensor=false&key=" & "          --- Google API Key ---          ", False

    Request.send
    Results.LoadXML Request.responseText
    Set Stat = Results.SelectSingleNode("//status")

    Select Case UCase(Stat.Text)

        Case "OK"
            Set Lat = Results.SelectSingleNode("//result/geometry/location/lat")
            Set Lon = Results.SelectSingleNode("//result/geometry/location/lng")
            GetGeo = Lat.Text & ";" & Lon.Text

        Case Else
            GetGeo = "Error"

    End Select

fehler:
    Set Stat = Nothing
    Set Lat = Nothing
    Set Lon = Nothing
    Set Results = Nothing
    Set Request = Nothing

End Function
```

Abbildung 4.22: Visual Basic Code für VBR-Editor Modul „Geocoding von Geokoordinaten“

Mithilfe der Google-API lassen sich so ein Großteil der fehlenden Geokoordinaten rekonstruieren. Dazu wird die ausgewählte Adresszeile eingelesen und eine Anfrage an die Google-API mit den Adressdaten geschickt. Das Antwortprotokoll in XML-Format wird ausgelesen und bei positivem Status die Längen- und Breitenkoordinate im Textfeldwert gespeichert. In vereinzelten Fällen kann anhand der vorhandenen Adressdaten kein Koordinatenbezug über die API erstellt werden. In diesen Fällen muss händisch nachkorrigiert werden. Sollte sich auch bei der händischen Suche kein Ergebnis feststellen lassen, kann über die „Postleitzahlenmethode“ ein Näherungswert für die Geokoordinate erstellt werden.

Natürlich gibt es noch weitaus mehr Szenarien, in denen fehlende Attributeinträge auf anderem Wege ergänzt werden müssen. Einige Angaben können nur aus dem Datenkontext selber ergänzt werden. So wie das in Abbildung 4.15 angeführte Beispiel. Die Angaben zur Treibstoffart „EngineGas“ können nicht automatisiert abgeglichen werden, da es keine zusätzlichen Fahrzeuginformationen zur Motorisierung oder dem Fahrzeugtypen gibt. Lediglich Angaben zum Fahrzeuggewicht und der Beladungsmenge sind verfügbar. Über diese kann jetzt versucht werden, eine bedingte Ableitung der Treibstoffart anhand von Abschätzungen zu bekommen. So kann man davon ausgehen, dass es sich bei einem Nettogewicht >= 2500 kg um einen LKW handelt und dieser mit Diesel betrieben wird. In diesem Fall lassen sich zwar nicht alle fehlenden Werte ergänzen, jedoch ein relativ hoher prozentualer Anteil. Nach Anwenden dieser Bedingung konnte die Anzahl von Null-Werten im Attribut „EngineGas“ von 23 auf 5 reduziert werden, wobei zwei der verbleibenden Null-Werte weitere Null-Werte in anderen Attributen aufweisen. Damit konnte die Null-Wert-Menge von 23,5% auf 5,75% gesenkt werden.

Nachdem die Null-Wert-Menge erfolgreich verbessert werden konnte, wird sich im Anschluss den Duplikaten angenommen. Obwohl Excel zur Duplikatprüfung eine direkte Funktion implementiert hat, ist eine effiziente Duplikatsuche, gerade bei textuellen Daten, durch unzählige Faktoren bedingt. Da die Duplikatsuche lediglich die in Zellen enthaltenen Werte bzw. Strings miteinander eins zu eins vergleicht, führt schon ein zusätzliches Zeichen oder eine Abweichung der Darstellung zu Fehlern in der Dupliaterkennung. Die anfängliche Konvertierung der Einträge der Datensätze ermöglicht dennoch eine relativ zuverlässige Erkennung von Duplikaten, indem Darstellungsform und verwendete Trennzeichen vereinheitlicht wurden, wie die folgenden Abbildungen 4.23 und 4.24 zeigen.

2499		1.211.469		L. ARTEMIDOS 92 ARTEMIS 19016 ARTEMIS	19016	37.9770421;24.0084151
2500	121.1469.1			L. ARTEMIDOS 92 ARTEMIS 19016 ARTEMIS	19016	37.9770421;24.0084151
2501		1.211.470		AL. ASIMAKOPOYLOY 25 AGIOS DIMITRIOS 17342 AGIOS DIMITRIOS	17342	37.9213369;23.7303362
2502		1.211.471		CHAROKOPOY 126 KALLITHEA 17676 KALLITHEA	17676	37.9618634;23.7060990
2503		1.211.472		YMITTOY 141 ATHENSPAGKRATI 11633 ATHENS	11633	37.9657929;23.7473600
2504		1.211.473		L. MESOGEION 154 (ENTOS NOS. GENNIMATA) ATHENS 11527 ATHENS	11527	37.9947378;23.7781256
2505		1.211.474		N.PLASTIRA 35 AIGALEO 12241 AIGALEO	12241	37.9942943;23.6823510
2506		1.211.475		KREMOY 26 KALLITHEA 17672 KALLITHEA	17672	37.9619439;23.7054423
2507	121.373.1.2			PLATEIA POLITEIAS KIFISIA KIFISIA	14563	39.6211;19.9225
2508	121.373.1.3			PLATEIA POLITEIAS KIFISIA 14563 KIFISIA	14563	38.0817;23.8302

Abbildung 4.23: Nodes Datensatz Duplikatsuche nach Attribut „Address“ ohne Normalisierung

Der „Nodes“ Datensatz enthält 7702 Dateneinträge zu „unterschiedlichen“ Standorten. Davon wurden ohne vorangehende Adresszeilennormalisierung 430 eindeutige Duplikate entdeckt. Das heißt, im Datensatz wurde schon an dieser Stelle mit einem einfachen 1:1 Abgleich des Attributs eine Fehlermenge von ~5,6% nachgewiesen.

2499		1.211.469		L. ARTEMIDOS 92 ARTEMIS	19016	37.9770421;24.0084151
2500	121.1469.1			L. ARTEMIDOS 92 ARTEMIS	19016	37.9770421;24.0084151
2501		1.211.470		AL. ASIMAKOPOYLOY 25 AGIOS DIMITRIOS	17342	37.9213369;23.7303362
2502		1.211.471		CHAROKOPOY 126 KALLITHEA	17676	37.9618634;23.7060990
2503		1.211.472		YMITTOY 141 ATHENSPAGKRATI ATHENS	11633	37.9657929;23.7473600
2504		1.211.473		L. MESOGEION 154 (ENTOS NOS. GENNIMATA) ATHENS	11527	37.9947378;23.7781256
2505		1.211.474		N.PLASTIRA 35 AIGALEO	12241	37.9942943;23.6823510
2506		1.211.475		KREMOY 26 KALLITHEA	17672	37.9619439;23.7054423
2507	121.373.1.2			PLATEIA POLITEIAS KIFISIA	14563	39.6211;19.9225
2508	121.373.1.3			PLATEIA POLITEIAS KIFISIA	14563	38.0817;23.8302

Abbildung 4.24: Nodes Datensatz Duplikatsuche nach Attribut „Address“ mit Normalisierung

Durch die zuvor angewandte Normalisierung wird die gefundene Duplikatanzahl deutlich gesteigert. Nachdem unterschiedliche Darstellungen der Daten größtenteils ausgeschlossen wurden, wird die gefundene Anzahl an eindeutigen Duplikaten auf 645 gesteigert. Das heißt, nur mittels Normalisierung wird die Genauigkeit der Duplikatsuche in diesem Fall um 50% erhöht. Damit enthält der Nodes Datensatz bisher 8,4% Duplikate.

Dazu kommen Duplikate, die durch Tippfehler oder Zeichenfehler nicht gefunden werden konnten. Diese wurden zum Teil durch einen zusätzlichen Abgleich über andere Attributsuchen gefunden. So konnten neben neuen möglichen Duplikaten auch Ergänzungen zu bereits gefundenen Duplikaten ermittelt werden.

2525	MPAKNANA 24 N.KOSMOS	11745	37.954;23.7229
2526	MPAKNANA 24 NEOS KOSMOS ATHENSGR	11745	37.954;23.7229

Abbildung 4.25: Nodes Datensatz Duplikatsuche nach Attribut „Address“ (rot) und „Coordinates“ (grün)

Die gefundene Duplikate, die in allen Attributen eindeutig übereinstimmen, können direkt gelöscht werden, sodass nur noch eine Version des Eintrags vorhanden ist. Bei fehlender Übereinstimmung in einzelnen Attributen wird überprüft ob die Abweichung durch einen fehlenden Eintrag entsteht oder dadurch, dass die Einträge gänzlich unterschiedlich sind.

Um nun gefundene Duplikate zusammen zu führen, damit nur eindeutige Einträge im Datensatz verbleiben, müssen die Dateneinträge zunächst nach dem Attribut sortiert werden, in dem Duplikate aufgetreten sind. Dadurch muss im Folgenden nicht jede Zeile mit jeder anderen verglichen werden, sondern nur benachbarte Zeilen werden paarweise verglichen. Im Anschluss wird geprüft, ob das Vergleichsattribut gleich ist. Ist dies der Fall, werden die Zeilen als Duplikate behandelt und zusammengeführt. Dazu werden die übrigen Attribute verglichen. Bei Gleichheit werden sie ebenfalls zusammengeführt. Enthält eine der beiden aktuell im Vergleich befindlichen Zeilen ein Null-Attribut oder einen Dummy-Wert, wird der entsprechende Wert der anderen Zeile für die Ergebniszeile verwendet.

Für das Beispiel des „Nodes“-Datensatzes heißt das, dass zunächst nach dem Attribut „Address“ sortiert wird und im Anschluss die Zeilen paarweise von unten nach oben verglichen werden. Übereinstimmende Attribute werden direkt übernommen, Abweichungen werden gesondert behandelt. Abweichungen bedingt durch Null- oder Dummy-Werte werden wie beschrieben behandelt. Unterschiede in Angaben wie der Geokoordinate („Coordinates“) werden sicherheitshalber erneut über die Google-API abgeglichen. Bei textuellen unformatierten Einträgen, wie der Beschreibung („Description“), wird direkt die Zeile entnommen, die mehr Informationen enthält. Alternativ kann für solche Daten auch die Wortmenge gebildet werden und im Zielattribut zusammengefügt werden um Dopplungen zu vermeiden.

Dieser Prozess kann mithilfe eines Visual Basic Moduls umgesetzt werden. Abbildung 4.26 zeigt den Quellcode für das Modul zur Duplikatzusammenführung des „Nodes“-Datensatzes.

```
Sub Duplikate()
Dim lngRow As Long
With ActiveSheet
'Definition des zu bearbeitenden Bereichs
lngRow = .Cells(7702, 3).End(xlUp).Row
.Cells(1).CurrentRegion.Sort key1:=.Cells(3), Header:=xlYes
    Do
    'Prüfe ob die Adresse der benachbarten Zellen gleich ist
        If .Cells(lngRow, 3) = .Cells(lngRow - 1, 3) Then
        'Vergleich der verbleibenden Attribute
            If .Cells(lngRow, 4) = .Cells(lngRow - 1, 4) Then
                .Cells(lngRow - 1, 4) = .Cells(lngRow, 4)
            ElseIf .Cells(lngRow, 4) = "" Then
                .Cells(lngRow - 1, 4) = .Cells(lngRow - 1, 4)
            Else
                .Cells(lngRow - 1, 4) = .Cells(lngRow, 4)
            End If

            If .Cells(lngRow, 5) = .Cells(lngRow - 1, 5) Then
                .Cells(lngRow - 1, 5) = .Cells(lngRow, 5)
            Else
        'Aufruf des Moduls zur Geokoordinaten Bestimmung
                .Cells(lngRow - 1, 5) = Modul3.GetGeo(.Cells(lngRow, 3))
            End If

        .Cells(lngRow - 1, 6) = .Cells(lngRow - 1, 6)
        .Cells(lngRow - 1, 7) = .Cells(lngRow - 1, 7)
    'Löschen des Duplikats
        .Rows(lngRow).Delete
        End If

        lngRow = lngRow - 1
    Loop Until lngRow = 2
    End With
End Sub
```

Abbildung 4.26: Quellcode zur Duplikatzusamenführung des „Node“-Datensatzes

Damit ist die Validierung an dieser Stelle abgeschlossen. Das Ergebnis der Validierung lautet, dass zu allen zuvor vorgestellten Fehlertypen vorhandene Fehler in den Praxis-Datensätzen gefunden werden konnten. Ebenso konnten alle Fehlertypen im Verlauf der Validierung behandelt und teils vollständig bereinigt werden. Die Anwendbarkeit des Modells wurde also überprüft. Da das Modell jedoch auf Excel-Datensätze zugeschnitten wurde, war das Ergebnis zu erwarten. Des Weiteren war die Fehlerkorrektur bei einer Vielzahl von Fehlern vollständig und zufriedenstellend. Ebenfalls ließen sich Daten die nicht vorhanden waren durch Näherungen reproduzieren. Das zeigt, dass eine grundlegende Anreicherung der lückenhaften Daten ermöglicht wurde. Duplikate in den Beispieldatensätzen konnten mithilfe der angewandten Methoden ebenfalls gekennzeichnet werden und in einem moderaten Maße zu eindeutigen Dateneinträgen zusammengefasst werden.

Insgesamt stellt das hier ausgearbeitete Konzept zur Bereinigung von Rohdatensätzen eine gute Basis für die Vorverarbeitung von Datensätzen im Excel-Datenformat dar.

4.2.3 Zusammenfassung und Fazit

Ziel der Arbeit war es, ein Konzept zu erarbeiten, mit dessen Hilfe es möglich ist, Datensätze in ihrem bestehenden Datenformat so vor zu verarbeiten, dass das Fehlermaß der Rohdaten möglichst stark gesenkt werden kann. Dadurch wird eine verbesserte Integration in Supply-Chain-Systeme gewährleistet. Die anschließende Validierung zeigt, dass das Ziel im erwarteten Rahmen erfüllt wurde und verbleibenden Datenprobleme und Fehlerquellen lokalisiert werden konnten.

Als Ausgangsbasis für diese Ausarbeitung wurde in Kapitel 2 ein Grundverständnis über Supply-Chain-Daten und die Funktion sowie Aufbau von Supply Chain mittels Literaturrecherche erarbeitet. Dadurch wurde zusätzlich die Notwendigkeit von Fehlerfreiheit von Supply-Chain-Daten aufgezeigt. Das von Ziegler (2015) erarbeitete Modell zur Kategorisierung von Supply-Chain-Daten ermöglichte es zudem, im Verlauf der Konzepterstellung auftretende Daten zur vereinfachten Verarbeitung zusammenzufassen. Hier konnte im Folgenden auf Datenqualität hemmende Faktoren eingegangen werden.

Die im dritten Kapitel getätigte Recherche zu Dateninkonsistenzen zeigte die zu fokussierenden Punkte der Datenbereinigung auf. Hier wurden sowohl Kernpunkte der Datenheterogenität, als auch die verschiedenen Datenfehlertypen von Eingangsdaten beschrieben. Dadurch wurde nötiges Grundverständnis zu den verschiedenen Fehlerklassen geschaffen, dieses wurde im Verlauf der Arbeit erneut aufgegriffen. Die folgenden Themen des Abschnitts 3.2 eröffneten die Thematik des eigentlichen Kernpunkts dieser Arbeit. Die beschriebenen Konzepte zur Heterogenitätsüberwindung und zur Fehlerbereinigung gaben eine Übersicht über sowohl theoretische als auch praktische Ansätze zur Datenbereinigung. Mithilfe der hier vorgestellten Ansätze konnte das Grundkonzept zur Bereinigung von Supply-Chain-Daten skizziert werden.

Hierfür wurde in Kapitel 4 ein theoretisches Ablaufmodell zur Vorbereinigung von Supply-Chain-Daten vorgestellt. Dazu wurden insbesondere die Ergebnisse aus Kapitel 2 und 3 kombiniert. Durch die in Abschnitt 2.2 angesprochenen Umstände der Datenbereitstellung, musste der Konzeptentwurf auf die Nutzung mit üblichen Datenformaten angepasst werden. Eine Erarbeitung eines Modells basierend auf der Vorverarbeitung von Excel-Daten war die Folge. Der Ablauf des Modells stützt sich auf das Vorgehen der zuvor in Abschnitt 3.2 beschriebenen Abläufe. Ergebnis der Arbeit ist ein Konzept, dass Ansätze zur Überwindung von auftretender Datenheterogenität in Microsoft Excel Datensätzen liefert, in dem bekannte Datenfehlerklassen in Datensätzen lokalisiert und Konzepte zur Fehlerkorrektur in Abhängigkeit der Datenkategorien vorgegeben werden.

Die praktische Anwendung des theoretisch erarbeiteten Konzeptes zur Vorbereinigung von Datensätzen zur Nutzung in Supply-Chain-Systemen zeigte, dass sich die Datenqualität der Beispieldatensätze durch die strukturelle Anwendung der einzelnen Schritte teils deutlich steigern ließ.

Die Validierung zeigte jedoch auch, dass die Konzepte auf die Daten der Datensätze angepasst werden müssen, um zuverlässige und zufriedenstellende Ergebnisse liefern zu können. Die Konvertierung und Harmonisierung war zwar zu einem relativ hohen prozentualen Anteil erfolgreich und konnte gerade bei der Rekonstruktion von Maßeinheiten über den Kontext anderer Dateneinträge gute Ergebnisse erzielen, jedoch war dieses Vorgehen bei stark von Transpositionen und Eingabefehlern belasteten Daten kein erfolgversprechender Ansatz. Oftmals war ebenfalls keine automatisierte Korrektur von Dateninkonsistenzen möglich, sondern händisches Bearbeiten notwendig, um die Daten in das gewünschte Format zu bringen oder Datenfehler auszugleichen. Datenfehler, die Stammdaten betreffen, wie Geokoordinaten oder Adresssätze konnten mit zufriedenstellender Zuverlässigkeit bereinigt und ergänzt werden.

Wie praktikabel das Vorgehen zur Ermittlung von Näherungswerten von geografischen Informationen in einem echten Supply-Chain-Umfeld ist, kann an dieser Stelle nur gemutmaßt werden. Dieser Sachverhalt müsste innerhalb echter Simulationen nachgestellt und die Güte der Näherungen bestimmt werden.

Ebenso blieben Laufzeiten von Anwendungen zur Bereinigung von Fehlern in einem Dokument unbeachtet. Die Nutzbarkeit muss demnach auch aus Sicht der Zeit- und Ressourceneffizienz betrachtet werden.

Abschließend hat sich im Verlauf des Prozesses gezeigt, dass die Verarbeitung stark unvollständiger Datensätze in einigen Fällen an ihre Grenzen stößt und sich einige Daten im Zuge dieses Konzeptes nicht ermitteln ließen. So waren Schlüsselwerte, wie fehlende EANs und Produkt-IDs nicht mit angewandten Methoden rekonstruierbar. Aus diesem Grund muss vor einer praktischen Verwendung des Konzepts geklärt werden, wie mit stark beschädigten Datensätzen verfahren wird und welche Auswirkung der Datenzustand auf deren Nutzen hat.

Auf dieser Arbeit basierende Arbeiten könnten das Modell zur Datenbereinigung durch neue Ansätze weiter verbessern, um auch die bisher ungelösten Datenfehler zu korrigieren. Eine Möglichkeit dazu wäre zum Beispiel die Nutzung von „Fuzzy Logic“- Algorithmen zu verbesserter Erkennung von Datenduplikaten.

5 Literaturverzeichnis

Literaturquellen:

[HoNo09] E. Hoffmann, F. Nothardt: Logistics Due Diligene – Analyse – Bewertung – Anlässe – Checklisten, Berlin, Springer Verlag 2009

[Hin02] H. Hinrichs: Datenqualitätsmanagement in Data Warehouse-Systemen, Universität Oldenburg, 2002

[Las06] W. Lassmann: Wirtschaftsinformatik. Nachschlagewerk für Studium und Praxis. 1. Auflage, Wiesbaden, Gabler Verlag, 2006

[Pir11] A. Prio, M. Gebauer: Definition von Datenarten zur Kommunikation im Unternehmen, entnommen aus: K. Hildebrand, M. Gebauer, H. Hinrichs, M. Mielke: Daten- und Informationsqualität. Auf dem Weg zur Information Excellence. 2. Auflage, Wiesbaden, Vieweg + Teubner Verlag, 2011

[Zie15] J. Ziegler: Systematische Untersuchung von möglichen Datenkategorien in Supply Chains, Technische Universität Dortmund, 2015

[Hil04] K. Hildebran: Datenqualität im Supply Chain Management, Fachhochschule Darmstadt, 2004

[Inm05] W. H. Inmon: Building the Date-Warehouse, 4th Edition, Indianapolis Indiana, Wiley Publishing, Inc., 2005

[Wro05] M. Wrobel: Multidimensionale, heterogene, visualisierbare Datenräume. Anforderungen, Entwurf und Implementierung einer adaptiven und interaktiven Schnittstelle für transdisziplinäre wissenschaftliche Daten im Kontext der Erdsystemanalyse, Freie Universität Berlin, 2005

[Mül13] R. M. Müller, H.-J. Lenz: Business Intelligence, Berlin Heidelberg, Springer Verlag, 2013

[Ven15] A. Vennemann: Vorgehensweise zur Aufbereitung von Eingangs- und Ergebnisdaten einer ereignisdiskreten Simulation eines Logistiknetzwerks des Werkstoffhandels zur glaubwürdigen Messbarkeit von komplexen Data-Warehouse-Kennzahlen, Technische Universität Dortmund, 2015

[KeFi98] H.-G. Kemper, R. Finger: Datentransformation im Data Warehouse, Konzeptionelle Überlegung zur Filterung, Harmonisierung, Verdichtung und Anreicherung operativer Datenbestände, entnommen aus: P. Chamoni, P. Gluchowski: Analyische Informationssysteme, Berlin Heidelberg, Springer Verlag, 1999

[Schna04] J. O. Schnabel: Formen der Heterogenität, Technische Universität Kaiserslautern, 2004

[LeNa07] U. Leser, F. Naumann: Informationsintegration, Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen

[RaDo00] E. Rahm, H. H. Do: Data Cleaning. Problems and Current Approaches, Universität Leipzig, 2000

[Nau07] F. Naumann: Datenqualität, Informatik Spektrum, Springer Verlag, 2007

[PloS05] J. Van den Broeck, S. A. Cunningham, R. Eeckels, K. Herbst: Data Cleaning: Detecting, and Editing Data Abnormalities, PLoS Medicine, 2005

Onlinequellen

[IDC17] IBM : IDC Manufacturing Insights: The Path of the thinking Supply Chain 2017 http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=WHW12345USEN& zuletzt geprüft 19.03.2018

[Bol01] Lexikon der Kartografie und Geomatik: Eigenschaft, Heidelberg: Spektrum Akademischer Verlag, 2001, http://www.spektrum.de/lexikon/kartographie-geomatik/eigenschaft/1128, zuletzt geprüft 19.03.2018

[Vah1] Prof. Dr. R. Vahrenkamp, Dr. C. Sieperman: Enterprise-Resource-Planing-System, Gabler Wirtschaftslexikon, Springer Verlag, http://wirtschaftslexikon.gabler.de/Archiv/17984/enterprise-resource-planning-system-v13.html, zuletzt geprüft 19.03.2018

[Wik1] Wikipedia: Material Requirements Planning, https://de.wikipedia.org/wiki/Material_Requirements/Planning, zuletzt geprüft 19.03.2018

[COW] M. Bayer: Computerwoche: Integrierte Planung Fehlanzeige, Excel bleibt das beliebteste Planungs-Tool, 2016, https://www.computerwoche.de/a/excel-bleibt-das-beliebteste-planungs-tool,3324960

6 Anhang

```
Sub AddressManipult()
Dim obj As Object
Dim C As Range
Dim arr() As String
Dim i As Integer
Dim boo As Boolean
Dim Key As Variant

For Each C In Selection
    'Aufruf der Funktionen "rep()" und "cut()"
    C.Value = rep(C)
    C.Value = cut(C)
    'Teilen des Eingabestrings nach Leerzeichen
    arr = Split(C.Value, " ")
    'Anlegen eines Scripting Dictionarys
    Set obj = CreateObject("Scripting.Dictionary")

    For i = 0 To UBound(arr)
    'Entfernen von Telefonnummern und Postleitzahlen
    If IsNumeric(arr(i)) And Len(arr(i)) > 4 Then
        boo = True
    Else
    End If
        If boo = False Then
    'Trimmen der überflüssigen Leerzeichen
        obj(Trim(arr(i))) = 0
        End If
    boo = False
    Next

    'Entfernen von "falschen" Einträgen aus dem Scripting Dictionary
    If obj.Exists(" ") Then obj.Remove " "
    If obj.Exists("") Then obj.Remove ""
    'Zusammensetzen des Strings
    C.Value = Join(obj.Keys, " ")
Next
End Sub
```

Abbildung 6.1: Quellcode VBA Modul zur Adresssatzbearbeitung

```
Function cut(rng As Range) As String
Dim tmp As String
Dim i As Integer
tmp = rng.Value

    'Trennen von Zahlen und Zeichen durch Leerzeichen
    For i = Len(tmp) - 1 To 1 Step -1
        If IsNumeric(Mid(tmp, i, 1)) <> IsNumeric(Mid(tmp, i + 1, 1)) Then
            tmp = Left$(tmp, i) & " " & Right(tmp, Len(tmp) - i)
        End If
    Next i

cut = tmp
End Function
```

Abbildung 6.2: Quellcode der Function „cut()“

```
Function rep(rng As Range) As String
Dim i As Integer
Dim arr As Variant
arr = Array(":", "(", ")", "+", ",", "/", "T.K.", "T.K", "TK.", "TK")

    'Entfernen von ungewollten Zeichen und Bezeichnern
    For i = 0 To UBound(arr)
    rng.Replace arr(i), ""
    Next i
rep = rng
End Function
```

Abbildung 6.3: Quellcode der Function „rep()“

Eidesstattliche Versicherung

_____ _____

Name, Vorname Matr.-Nr.

Ich versichere hiermit an Eides statt, dass ich die vorliegende Bachelorarbeit/Masterarbeit* mit dem Titel

_____

_____

_____

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

_____ _____

Ort, Datum Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -)

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird gfls. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen:

_____ _____

Ort, Datum Unterschrift