1. Schwer zu entdecken: wenn Datensätze fehlen
2. Data Profiling . Spalten prüfen
3. branchenabhängig
4. Eingaben sicherstellen
5. Statistik
6. Integration
7. Notiere Beispiele von Problemen der Datenqualität die eher schwer zu entdecken sind.

 **Fehlende komplette Datensätze – nur mit fachlichem Wissen prüfbar**

 **Semantische Inkonsistenzen**: Ein Feld „Status“ kann in einer Quelle „active/inactive“ und in einer anderen „1/0“ meinen – schwer zu erkennen ohne Kontext.

 **Veraltete Daten**: Daten sind formal korrekt, aber nicht mehr aktuell (z. B. alte Telefonnummern oder Adressen).

 **Unplausible Kombinationen**: Z. B. ein Kunde aus „Berlin“ mit einer Postleitzahl aus „München“ – das ist nicht immer offensichtlich.

 **Verdeckte Dubletten**: Gleicher Kunde mit leicht unterschiedlichem Namen („M. Müller“ vs. „Max Müller“).

 **Verletzung fachlicher Regeln**: Z. B. Kinder unter 18 mit Kreditkarten – technisch okay, aber inhaltlich zweifelhaft.

** Falsche Angaben**

** Falsche Initialwerte: Bsp: eine Tuere sendet Binaerdaten**

falsche Rundung

andere Sprache

1. Erkunde den Datenbestand der Datei customer.csv. Wie gehst Du vor? Was fällt Dir auf? ZIEL:FEHLERARTEN FINDEN

Folgende Felder sind nicht normalisiert: contact\_name, address. In Spalte L fehlt die Überschrift, es könnte sich um eine Telefonnummer handeln, es gibt aber auch Faxnummern. Teilweise fehlen Angaben, offenbar waren es keine Musseingabefelder.

Wie geht’s Du vor?

Man muss normalisieren (zb für contact\_name; nicht Max Müller sondern Max und Müller in eigenen Spalten.

Musterdatenstruktur festlegen und die Datentypen definieren; mittels Data Dictionary definieren.

**Spalte Contact\_name**

-missing values

**Spalte**

Was fällt Dir auf?

Die Datenqualität ist schlecht.

Es gibt Messy code, das konnte nicht geparsed werden.

1. Welche Folgen können eine schlechte Datenqualität haben?

Eine schlechte Datenqualität kann folgende Folgen haben:

Es können keine oder nur sehr vage Aussagen über die Inhalte getätigt werden, somit fehlt die Entscheidungsgrundlage. Wenn nicht ausreichend transparent ist, dass die Datenqualität nicht ausreichend oder zu schlecht ist, kann dies zu Fehlinterpretationen führen. Und indirekt zu Imageschaden, Umsatzrückgang und Verlust von Kunden.

Fehlinformation von Kunden und erhöhte Marketing/Vertriebskosten.

Mehraufwand bei Korrekturen

Integrationsprobleme mit anderen Systemen

1. Wo setzt Du an, um die Datenqualität langfristig zu verbessern? Bitte nenne Beispiele.

Aus dem Data Profiling sollte man die Erkenntnisse in Assessment und Monitoring einfließen lassen.

Man sollte bereits bei der Konzeption des Systems sicherstellen, dass die Dateneingabe nur strukturiert, für die richtigen Datentypen und bestmöglich mit einer Gültigkeitsprüfung versehen ist, sodass Fehleingaben erst gar nicht möglich sind. Über die Steuerung von Kann und Mussfeldern kann sichergestellt werden, dass notwendige Informationen eingegeben werden müssen und somit logische Abhängigkeiten sichergestellt werden können. (Assessment)

Durch Prozessdokumentationen kann den Anwendern veranschaulicht werden, wie welche Daten an welcher Stelle korrekt eingegeben werden können. (Monitoring)

1. Du möchtest einen Datensatz (siehe die Dateien customer.csv sowie orders.csv) hinsichtlich der Datenqualität begutachten.
2. Macht der Einsatz beschreibender Verfahren der Statistik hierbei Sinn?

-ja, mit Statistik kann man auswerten wie die Datenqualität der Quellendaten ist.

1. Falls ja, welche Verfahren kannst Du bei customer.csv sowie orders.csv zum Einsatz bringen?

Z-score verfahren, für die Spalte wie Adresse sollte man regex von Musterdaten einsetzen, danach mit stat. Die Fehlerdatenverteilung analysieren.

Häufigkeitstabellen: z. B. Kunden nach Ländern oder Städten

Verteilungsanalyse: z. B. Bestellungen pro Mitarbeiter

Fehlwertanalyse

Deskriptive Kennzahlen (Mittelwert, Median, Ausreißer in employee\_id etc.)

1. Von was hängt deren Einsatz ab?

Datenmodel, Datenmenge, Definition der Abnormaldaten

Art der Daten

Ziel der Analyse

Vorhandene Tools und Datenmenge

1. Welche spezifischen Probleme können bei der Integration aus mehreren Datenquellen auftauchen? Auf was ist hierbei ggf. zu achten?

**Spezifische Probleme können auftauchen bei:**

* Inkonsistente Datenformate
* Abweichende Datentypen
* Namenskonflikte / uneinheitliche Bezeichnungen
* Fehlende Schlüssel oder keine gemeinsamen Schlüssel
* Semantische Unterschiede
* Redundanzen und Dubletten
* Unterschiedliche Granularität
* Unterschiedliche Aktualität
* Unterschiedliche Vorgaben und unterschiedliches Verständnis der Datenbasis
* Unklare oder fehlende Dokumentation

1. Weshalb ist es von Vorteil ein Data Dictionary für die betreffenden Daten zu haben?

* Einheitliches Verständnis der Daten
* Unterstützung bei Datenintegration
* Erleichtert die Datenanalyse und -qualitätssicherung
* Grundlage für automatisierte Checks und Validierungen
* Unterstützt Data Governance & Compliance
* Bessere Dokumentation und Wissensweitergabe
* Nachvollziehbarkeit

1. Inwiefern ist die Entstehung bzw. die Erhebung der Daten aus Sicht der Datenqualität von Relevanz?

Eine fehlerhafte Erfassung führt zu schlechten Basisdaten. Unklare Erhebungsprozesse führen zu Inkonsistenzen. Mittels automatisierter Eingabe können im Vergleich zu manueller Erfassung positive Qualitätseffekte erzielt werden. Eine fehlerfreie Datenbasis minimiert Fehleranfälligkeiten im weiteren Datenverarbeitungsprozess.

1. Inwiefern können Data Understanding (Datenverständnis) und Datenqualität zusammenhängen?

Ein gutes Datenverständnis ist entscheidend, um Probleme frühzeitig zu erkennen und schnell darauf reagieren zu können. Es ermöglicht zudem die gezielte Umsetzung von Qualitätsmaßnahmen, da Daten korrekt interpretiert und effektiv genutzt werden können. Darüber hinaus trägt ein solides Verständnis der Daten dazu bei, Fehlinterpretationen in Analysen zu vermeiden und somit fundierte Entscheidungen zu treffen.

Super Frage – die **Integration aus mehreren Datenquellen** ist ein klassischer Stolperstein in vielen Datenprojekten. Hier sind die **wichtigsten Probleme**, die dabei auftauchen können, inklusive **worauf man achten sollte**:

**🔧 Spezifische Probleme bei der Datenintegration**

1. **Inkonsistente Datenformate**
   * Beispiel: Datumsformate unterschiedlich (2025-04-23 vs. 23.04.2025)
   * **Achten auf**: Vereinheitlichung der Formate (Standardisierung, z. B. ISO 8601)
2. **Abweichende Datentypen**
   * Beispiel: In einer Quelle ist „Kundennummer“ ein String („000123“), in einer anderen ein Integer (123)
   * **Lösung**: Konvertieren in einen einheitlichen Typ, evtl. mit führenden Nullen auffüllen
3. **Namenskonflikte / uneinheitliche Bezeichnungen**
   * Beispiel: Spalte „id“ in einer Datei meint „Produkt-ID“, in einer anderen „Kunden-ID“
   * **Wichtig**: Eindeutige, sprechende Namen verwenden und dokumentieren
4. **Fehlende Schlüssel oder keine gemeinsamen Schlüssel**
   * Beispiel: Kunden in einem System nur mit E-Mail, im anderen mit Kundennummer
   * **Achten auf**: Aufbau von Mappings oder Übersetzungstabellen (z. B. Master Data Management)
5. **Semantische Unterschiede**
   * Beispiel: Feld „Status“ – in einem System „aktiv/inaktiv“, im anderen „1/0“, evtl. sogar andere Bedeutung
   * **Lösung**: Harmonisierung und semantisches Mapping
6. **Redundanzen und Dubletten**
   * Gleiche Daten mehrfach vorhanden mit leichten Abweichungen (z. B. „Max Mustermann“ vs. „M. Mustermann“)
   * **Beachten**: Dublettenabgleich, z. B. mit Fuzzy-Matching oder eindeutigen IDs
7. **Unterschiedliche Granularität**
   * Beispiel: Verkäufe auf Tagesbasis vs. monatlicher Bericht
   * **Aufpassen auf**: Aggregationslogik klar definieren
8. **Unvollständigkeit / Fehlende Werte**
   * Eine Quelle enthält Infos, die in der anderen komplett fehlen (z. B. Telefonnummern nur in CRM, nicht im Onlineshop)
   * **Lösung**: Datenanreicherung oder gezielte Ergänzung
9. **Unterschiedliche Aktualität**
   * Datenquelle A wird täglich aktualisiert, Quelle B nur einmal im Monat
   * **Wichtig**: Zeitstempel prüfen und Synchronisationsstrategien festlegen
10. **Unklare oder fehlende Dokumentation**

* Ohne klare Beschreibung der Daten ist eine sinnvolle Integration fast unmöglich
* **Tipp**: Aufbau eines **Data Dictionary** oder eines **Datenmodells**

**🧭 Worauf du achten solltest:**

* **Klare Zieldefinition**: Was willst du mit der Integration erreichen?
* **Quellen dokumentieren**: Herkunft, Format, Aktualität, Verantwortliche.
* **Validierungsregeln definieren**: Was ist gültig? Was nicht?
* **Metadaten sammeln**: Was bedeutet jede Spalte?
* **Automatisierung prüfen**: Regelmäßige Integration? Dann ETL/ELT-Prozesse aufsetzen.
* **Monitoring einrichten**: Qualität und Konsistenz regelmäßig überwachen.

Wenn du ein praktisches Beispiel brauchst, wie man zwei Datenquellen zusammenführt (z. B. customer.csv + orders.csv nach customer\_id), sag einfach Bescheid – ich zeig’s dir gern im Code oder Diagramm.

Ein **Data Dictionary** (auch **Datenkatalog** oder **Datenverzeichnis**) ist ein zentrales Werkzeug im Datenmanagement – und hat eine Menge Vorteile. Hier kommt eine klare und kompakte Übersicht, **warum** es so nützlich ist:

### ✅ ****Vorteile eines Data Dictionary****

#### **1. Einheitliches Verständnis der Daten**

* Es erklärt **klar und eindeutig**, was jede Spalte/Feld bedeutet.
* Beispiel: Was heißt „Status“? Aktueller Zustand des Kunden, der Bestellung – oder etwas ganz anderes?

🔍 **Fazit:** Vermeidet Missverständnisse zwischen IT, Analyse-Team und Fachabteilungen.

#### **2. Unterstützung bei Datenintegration**

* Beim Zusammenführen verschiedener Datenquellen hilft ein Data Dictionary, **gleichnamige, aber unterschiedlich gemeinte Felder zu erkennen** – oder umgekehrt.
* Beispiel: „ID“ in Quelle A ist die Kunden-ID, in Quelle B die Produkt-ID.

🔍 **Fazit:** Bessere Harmonisierung und Matching von Datenquellen.

#### **3. Erleichtert die Datenanalyse und -qualitätssicherung**

* Du kannst direkt sehen:
  + Welche Werte sind zulässig?
  + Welche Formate gelten?
  + Welche Felder dürfen **nicht leer** sein?

🔍 **Fazit:** Schnellere Fehlererkennung, bessere Kontrolle der Datenqualität.

#### **4. Grundlage für automatisierte Checks und Validierungen**

* Datenvalidierung lässt sich automatisieren, wenn z. B. im Dictionary steht:
  + birth\_date: Format = YYYY-MM-DD, darf nicht in der Zukunft liegen.

🔍 **Fazit:** Spart Zeit, reduziert manuelle Prüfungen.

#### **5. Unterstützt Data Governance & Compliance**

* Du weißt, **welche Daten sensibel** sind (z. B. E-Mail, Geburtsdatum).
* Wer ist **verantwortlich** für die Datenpflege („Data Owner“)?
* Was darf **verarbeitet oder exportiert** werden?

🔍 **Fazit:** Hilft bei Datenschutz (z. B. DSGVO) und Revisionssicherheit.

#### **6. Bessere Dokumentation und Wissensweitergabe**

* Neue Teammitglieder oder externe Partner können sich schneller einarbeiten.
* Kein „Tribal Knowledge“ mehr – Wissen wird dokumentiert, nicht nur in Köpfen.

🔍 **Fazit:** Fördert Teamarbeit und Kontinuität im Projekt.

### 📌 Typische Inhalte eines Data Dictionary

| **Feldname** | **Beschreibung** | **Datentyp** | **Format** | **Pflichtfeld** | **Wertebereich / Regeln** |
| --- | --- | --- | --- | --- | --- |
| customer\_id | Eindeutige Kundennummer | Integer | — | Ja | Nur positive Zahlen |
| gender | Geschlecht | String | „M“ / „F“ | Nein | M, F, divers, leer erlaubt |
| signup\_date | Datum der Registrierung | Date | YYYY-MM-DD | Ja | nicht in der Zukunft |

Wenn du willst, kann ich dir eine Vorlage oder ein Template für ein Data Dictionary geben – z. B. als Excel oder Markdown. Sag einfach Bescheid!