
Probleme aus der Praxis: Implementierung und Entwurf

Prof. Dr. Christian Pape

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

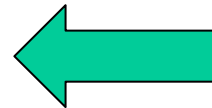
Integration Immobilienanwendung

- Eigenentwicklung für Kantonalbanken (Reimplementierung)
- Austausch Immobiliendaten zwischen Eigenentwicklung und ImmoScout24
- Dateiaustausch: Format Immoscout24 + FTP Server

Neu renovierte Liebhaberwohnung	
Staufferstrasse 12, 3006 Bern	
Das Wichtigste auf einen Blick	
Objektart	Wohnung
Preis	CHF 720'000.-
Garage	CHF 30'000.-
Anz. Zimmer	4
Wohnfläche	112 m ²
Stockwerk	2. OG
Anzahl Etagen im Gebäude	4
Baujahr	1924
Zustand	frisch renoviert
Bezug per	per sofort
ImmoCode	EYNJK
Objektnummer	1
Details zur Lage und Umgebung	
Autobahnanschluss	Bern-Ostring
Distanz zur Autobahn	1 km
Schulen im Ort	Sekundarschule, Primarschule, Kindergarten
Details zum Objekt	
Aussensitzplätze	- Anz. Balk - Anz. Gart - Anz. Terr
Nasszellen	- Anz. Bade - Anz. Bade - Anz. WCs
Weitere Räume	- Kellerräu
Parkplätze	- Tiefgarag
Waschküche	ja - Anz. Was - Anz. Tum
Heizung	Elektrospe
Anschlüsse	ISDN, Kab
Kontakt	
Herr	[Redacted]
Post	[Redacted]
6403 Küssnacht	



FTP



Komplett sanierte 2 1/2, 3 1/2 und 4 1/2 Zimmer-Eigentumswohnungen	
Hauptangaben	
Objektart	Wohnung, Andere
Adresse	Mädergutstrasse 35 3018 Bern
Kanton	Bern
Kaufpreis	CHF 262'000
Zimmer	3.5
Verfügbarkeit	nach Vereinbarung
Grössenangaben	
Baujahr	1971
Wohnfläche	88 m2
Innen	
Bodenbeläge	Parkett & Steinböden
Anzahl Bäder	1
Zustand	renoviert
hell	✓
ruhig	✓
sonnig	✓
Technik	



Integration Immobilienanwendung

■ Problem

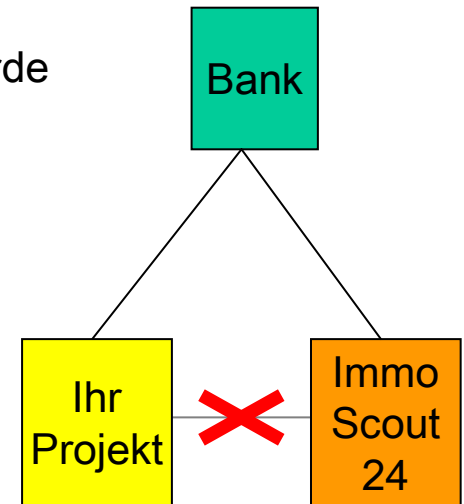
- Dateiformat von Immoscout24 lesen und erzeugen
- Es gab keine Kundenanforderungen (Schnittstellen wurde schlicht vergessen)
- Aber: **keine** Spezifikation dieses Formats vorhanden!
 - Syntax und Semantik der einzelnen Felder?, Zeichensatzcodierung?, Fluchtsymbole?
- Direkte Anfragen an Immoscout24 zwecklos, da **nur Vertragsverhältnis mit Bank** existierte
- Indirekte Anfragen waren nutzlos (Zeit, Qualität der Antworten)

■ Notlösung

- Viele **Beispieldateien** ansehen
- Alte Importschnittstelle ansehen (War eine Methode in einer Klasse mit ca. 120 einzelnen JDBC Anfragen)
- Viel Testen sinnvoll, aber während Entwicklung nicht möglich, da keine Vertragsverhältnis zu Immoscout24

■ Trotzdem

- Viele „Fehler“ traten erst Wochen nach Produktionsbeginn auf (Umlaute, HTML)



Integration Immobilienanwendung

- Dateiformat
 - Kein Standard, kein XML
 - Eigenentwicklung
 - Pro Immobilien eine **Zeile**
 - Werte durch # abgetrennt
 - Schlechte CSV Variante
 - Es existierte ein altes Excel von Vorgängerprojekt mit grober Definition der Felder (was sie ungefähr bedeuten), aber keine Definition der Syntax
- Felder enthielten z.B.
 - Namen, Telefon, Strasse, Ort, ...
 - URL von Abbildungen, Titeltext dieser Abbildungen
 - Beschreibungen der Objekte, Preis, ...
 - Texte konnten HTML sein (Besitzer der Immobilie tippte sie ein)

Manfred Mustermann#Villa am Strand#sehr schöne Aussicht, günstig#10000#

- Welche Probleme traten während Entwicklung und Produktion auf?

Integration Immobilienanwendung

■ Entwicklung

- Agregierte Informationen in der Datei
 - „Musterstrasse 5“, „Frau Marie Musterfrau“
 - wir benötigten zum Import in die Datenbank aber Strasse + Hausnummer getrennt
- Zuordnungsproblem bei Ortsnamen und Strassen:
 - Daten im Zielsystem waren aus offiziellen Schweizer Postverzeichnis (hohe Datenqualität)
 - Importierte Daten aber nicht

■ Produktion

- Export: Umlaute waren in Immoscout24 nicht korrekt (Zeichensatzcodierung war falsch)
- Export: Manche Informationen tauchten bei Immoscout24 nicht am richtigen Ort auf
- Import: Texte enthielten # als Zeichen, z.B.
 für einen Zeilenumbruch, aber es existierte kein Fluchtsymbol für #!

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

Agregierte Daten

- Agregierte Daten „deagregieren“
 - „Musterstrasse 5“ in Name der Strasse und Hausnummer zerlegen
 - „Fau Elise Exitus“ in Vorname (mit Anrede) und Nachnamen zerlegen
- Nicht einfach Draufloslegen
 - Liste mit vielen *unterschiedlichen* Beispielen von den existierenden Dateien machen

Agregierte Daten

■ Strassen

- „Musterstrasse 5“
- „Musterstrasse 5b“
- „Musterstrasse 5-10“
- „Musterstr. 17“
- „Am Pfad 21“
- „Weiher“
- „Postfach“
- „Postfach 43532“

■ Namen

- „Christian Pape“
- „Herr Christian Pape“
- „Herr Müller-Lüdenscheid“
- „Dr. Christian Pape“
- „Manfred Klaus Müller“
- „Prof. Dr. Christian Pape“
- „Dr. Pape“
- „Pape, Christian“
- „Seine Eminenz der Erzbischof von Köln“

Diese Fälle kamen vor.
Lösung notwendig.

Agregierte Daten: Namen

- Zieldatenbank kannte keine Anreden / Titel
 - Teil des Vornamens
- Wie erkenne ich den Nachnamen?
- Namen
 - „Christian Pape“
 - „Herr Christian Pape
 - „Herr Müller-Lüdenscheid“
 - „Dr. Christian Pape“
 - „Manfred Klaus Müller“

Lösung?

Aggregierte Daten: Namen

- Zieldatenbank kannte keine Anreden / Titel
 - Teil des Vornamens
- Wie erkenne ich den Nachnamen?
- Namen
 - „Christian Pape“
 - „Herr Christian Pape“
 - „Herr Müller-Lüdenscheid“
 - „Dr. Christian Pape“
 - „Manfred Klaus Müller“

Lösung?

- Zeichenkette aufspalten nach Leerzeichen
- Letzten Teil als Nachname nehmen, Rest als Vorname

Aggregierte Daten: Strassen

- Strassen
 - „Musterstrasse 5“
 - „Musterstrasse 5b“
 - „Musterstrasse 5-10“
 - „Musterstr. 17“
 - „Am Pfad 21“
 - „Am Pfad“
 - „Weiher“
- Woran erkenne ich den Strassennamen?
- Woran erkenne ich die Hausnummer?
- Eins von beiden reicht, dann hat man auch den Rest
- Es sollten möglichst viele Fälle *korrekt* erkannt werden

Lösung?

Agregierte Daten: Strassen

- Strassen
 - „Musterstrasse 5“
 - „Musterstrasse 5b“
 - „Musterstrasse 5-10“
 - „Musterstr. 17“
 - „Am Pfad 21“
 - „Am Pfad“
 - „Weiher“
- Woran erkenne ich den Strassennamen?
- Woran erkenne ich die Hausnummer?
- Eins von beiden reicht, dann hat man auch den Rest
- Es sollten möglichst viele Fälle *korrekt* erkannt werden

Lösung?

- Zeichenkette aufspalten nach Leerzeichen
- Letzten Teil von hinten wie folgt untersuchen
 - Leerzeichen eliminieren
 - Verhältnis Zahlen zu „nicht Zahlen“ bilden
 - Falls $v \geq 0.5$ ist, dann ist es eine Hausnummer

Agregierte Daten: Strassen

```
public String getHausnummer(String strasse) {
    String [] teileDerStrasse;
    String hausnummer = "";
    int digits = 0;

    strasse = strasse.trim();
    teileDerStrasse = strasse.split(" ");
    hausnummer = teileDerStrasse[teileDerStrasse.length-1].trim();

    for (char c : hausnummer.toCharArray() ) {
        if( Character.isDigit(c) ) {
            digits++;
        }
    }

    if (2 * digits >= hausnummer.length() ) {
        return hausnummer;
    } else {
        return "";
    }
}
```

Agregierte Daten: Strassen

- JUnit Testklasse für alle vorher aufgestellten Fälle implementieren

```
public void testGetHausnummer1() {
    assertEquals("5", strasse.getHausnummer("Musterstrasse 5") );
}

public void testGetHausnummer2() {
    assertEquals("5b", strasse.getHausnummer("Musterstrasse 5b") );
}

public void testGetHausnummer3() {
    assertEquals("5-10", strasse.getHausnummer("Musterstrasse 5-10") );
}

// 4,5,6 fehlen
public void testGetHausnummer7() {
    assertEquals("", strasse.getHausnummer("Weiher") );
}
```

Aggregierte Daten: Strassen

Lösung?

- Zeichenkette aufspalten nach Leerzeichen
- Letzten Teil von hinten wie folgt untersuchen
 - Leerzeichen eliminieren
 - Verhältnis v Zahlen zu „nicht Zahlen“ bilden
 - Falls $v \geq 0.5$ ist, dann ist es eine Hausnummer

Agregierte Daten: Strassen

- Die Lösung muss mit Kunden abgesprochen sein!
 - Die Lösung beschreibt **WAS** gemacht werden soll.
 - Sie ist damit Teil der **Anforderungen** des Systems.
- Falls Sie die Anforderungen **nicht** vom Kunden abnehmen lassen, passiert folgendes:
 - Nicht erkannte Hausnummern werden vom Kunden A ggf. als Softwarefehler behandelt und sie müssen diese während der Gewährleistung „umsonst“ beheben
 - Kurz nach Ihren Änderungen interpretiert Kunde B nun ihre Änderung als Fehler

Anforderung:

- Zeichenkette aufspalten nach Leerzeichen
- Letzten Teil von hinten wie folgt untersuchen
 - Leerzeichen eliminieren
 - Verhältnis v Zahlen zu „nicht Zahlen“ bilden
 - Falls $v \geq 0.5$ ist, dann ist es eine Hausnummer

Aggregierte Daten

- Aufgaben (könnte in der Klausur vorkommen)
 - Überlegen und Implementieren Sie jeweils eine Lösung für alle der aufgeführten Beispielfälle
 - Bei Namen
 - Wie bekommen Sie Anrede und Titel heraus?
 - Schweiz: Anrede, Titel kann deutsch, französisch oder italienisch sein (rätoromanisch ist zwar vierte offizielle Sprache, wird aber normalerweise nicht unterstützt)
 - Bei Orten
 - „76131 Karlsruhe“
 - Wie bekomme ich die Postleitzahl?
 - Was für Fälle gibt es alles?

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

Zuordnungsproblem

- Zieldatenbank (Eigenentwicklung)
 - Strassen, Ortsnamen, Postleitzahlen der kompletten Schweiz waren aus einem Postleitzahlenverzeichnis übernommen
 - Es durften keine neuen Einträge vorgenommen werden (Datenqualität)
 - Benutzereingaben + Importdaten mussten bestehenden Strassen, Orten zugeordnet werden
- „Anforderung“ vom Kunden
 - Soll so, „wie im alten System“ gelöst werden

Zuordnungsproblem

Reale Daten

- 3014 Bern-Wankdorf
- 3014 **Bern**
- 3014 **Berne**
- 3000 Bern
- 3014 BE-Wankdorf
- 3014 **Wankdorf**
- 3013 Bern-Breitenrain
- 3013 **Bern**
- 3512 Walkringen
- **9999** Walkringen
- 3512 **Wikartswil**



Postleitzahlenverzeichnis:

PLZ
3000
3014
3013
3512

PLZ	NAME
3000	Berne
3000	Bern
3014	Bern-Wankdorf
3013	Bern-Breitenrain
3512	Walkringen

Lösung?

- Best mögliche Datenqualität!
- Keine neuen Einträge schaffen!

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

Umlaute

- Umlaut bei Export nach Immoscout24 nicht vorhanden
- Offensichtlich in falschem Textformat exportiert
- Lösung: Über Kunden, nachfragen um welche Zeichensatzcodierung es sich handelt
- Antwort: ANSI (glaube ich)
 - Es gibt dutzenden ANSI Varianten
 - Vermutlich Windows-ANSI gemeint
 - Damit funktionierte es dann

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

Falsch zugeordnete Werte

- Beide Anwendungen haben unterschiedlichen Datenbanken
- Unterschiedliche Codierungen. Z.B. unterschiedliche Wohnarten
- Korrekte Abbildung der Werte und Codierungen nötig (Spezifikation)
- Da Zielformat nicht „bekannt“, ist dies natürlich nur eingeschränkt möglich

Neu renovierte Liebhaberwohnung	
Staufferstrasse 12, 3006 Bern	
Das Wichtigste auf einen Blick	
Objektart	Wohnung
Preis	CHF 720'000.-
Garage	CHF 30'000.-
Anz. Zimmer	4
Wohnfläche	112 m ²
Stockwerk	2. OG
Anzahl Etagen im Gebäude	4
Baujahr	1924
Zustand	frisch renoviert
Bezug per	per sofort
ImmoCode	EYNJK
Objektnummer	1
Details zur Lage und Umgebung	
Autobahnanschluss	Bern-Ostring
Distanz zur Autobahn	1 km
Schulen im Ort	Sekundarschule, Primarschule, Kindergarten
Details zum Objekt	
Aussensitzplätze	- Anz. Balk - Anz. Garte - Anz. Terrä
Nasszellen	- Anz. Bade - Anz. Bade - Anz. WCs
Weitere Räume	- Kellerräu
Parkplätze	- Tiefgarag
Waschküche	ja - Anz. Was - Anz. Tum
Heizung	Elektrospe
Anschlüsse	ISDN, Kab
Kontakt	
Herr	
Post	
6403 Küssnacht	

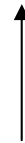
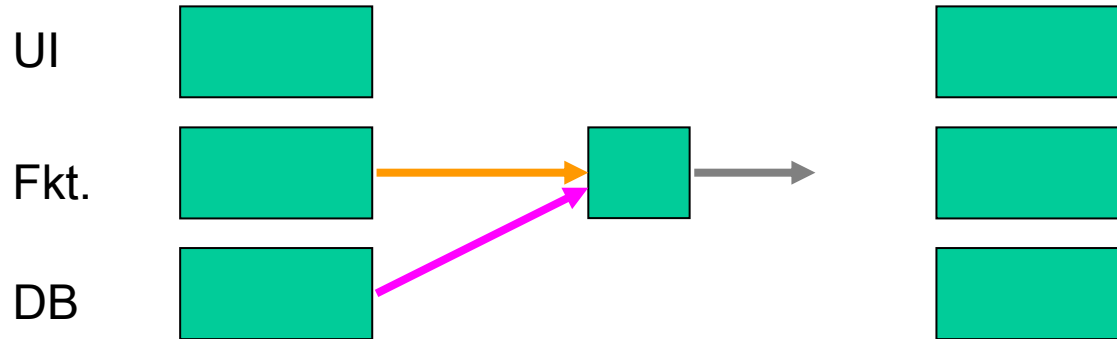


Abbildung
spezifizieren

Komplett sanierte 2 1/2, 3 1/2 und 4 1/2 Zimmer-Eigentumswohnungen	
Hauptangaben	
Objektart	Wohnung, Andere
Adresse	Mädlergutstrasse 35 3018 Bern
Kanton	Bern
Kaufpreis	CHF 262'000
Zimmer	3.5
Verfügbarkeit	nach Vereinbarung
Grössenangaben	
Baujahr	1971
Wohnfläche	88 m ²
Innen	
Bodenbeläge	Parkett & Steinböden
Anzahl Bäder	1
Zustand	renoviert
hell	✓
ruhig	✓
sonnig	✓
Technik	



Falsch zugeordnete Werte



- **Datenebene** (Datenbank zu Immoscout24 Format)
 - Abbildung von Feldern zu Feldern (Wo ist gleiche Information?)
 - Abbildung von Werten von Feldern zu Werten von Feldern (Welche Werte bedeuten das gleiche?)
 - ER-Model sinnvoll
- **Objektebene** (Getter-Methoden zu Immoscout24 Format)
 - Abbildung von Methoden einer Klasse zu Feldern (Wo ist gleiche Information?)
 - Abbildung von Werten/Klassen dieser Methode zu Werten von Feldern (Welche Werte bedeuten das gleiche?)
 - Klassendiagramm sinnvoll

Falsch zugeordnete Werte

- Abbildung tabellarisch z.B. mit „Excel“ kontinuierlich fortführen
- Objektebene (Getter-Methoden zu Immoscout24 Format)
 - Abbildung von Methoden einer Klasse zu Feldern (Wo ist gleiche Information?)
 - Abbildung von Werten/Klassen dieser Methode zu Werten von Feldern (Welche Werte bedeuten das gleiche?)

Klasse.Property / Typ	Bedeutung	Feld	Kommentar
Contact.FullName / string	Anrede, Vor- und Nachname der Kontaktpersn	#12	
Advertisement.ID / int	Eindeutige Identifikation der Anzeige	#18	Zielformat 10 Ziffern mit führenden 0en 0000087653
Subklassen von Room	Raumart	#60	
			Wo ist gleiche Information?

Falsch zugeordnete Werte

Welche Werte bedeuten das gleiche?

Im Extremfall: eine Tabelle pro Feld im Zielsystem
Hier: Beispiel Subklassen von Room

Wert / Einschränkung	Bedeutung	Wert von Feld #60	Kommentar
Balcony	Balkon	7	
Shower	Badezimmer nur mit Dusche	9	
Bath	Badezimmer mit Badewanne	9	
Garage / mit Garage.number = 1	Einzelgarage	10	Einzelgarage
Garage / mit Garage.number >= 2	Garage mit mehr als ein Stellplath	11	Doppelgarage (Code für mehr als zwei Stellplatz gibt es nicht)

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

Dateiformatsprobleme

- Texte enthielten # als Zeichen, z.B.
 für einen Zeilenumbruch, aber es existierte kein Fluchtsymbol für #!
- Lösungen
 1. a) Vor Aufsplitten einer Zeile in seine Felder nach „#“ erst alle nicht Feldtrenner # ersetzen durch ein Fluchtsymbol, das nicht # enthält
b) Nach Aufsplitten pro Feld Ersetzung wieder rückgängig machen
 2. Alle Feldtrenner # durch eine Zeichenfolge ersetzen, die sonst nicht vorkommt

```
Manfred Mustermann#Villa am Strand#sehr sch&ouml;ne &#10;Aussicht, <em>günstig</em>#10000#
```

1. a) &#n; in &n; ersetzen, color=# in color=

```
Manfred Mustermann#Villa am Strand#sehr sch&ouml;ne &#10;Aussicht, <em>günstig</em>#10000#
```

b) &n; in &#n; color= in color=# zurückübersetzen

```
sehr sch&ouml;ne &#10;Aussicht, <em>günstig</em>
```

2. Alle Feldtrenner # durch z.B. | ersetzen

```
Manfred Mustermann|Villa am Strand|sehr sch&ouml;ne &#10;Aussicht, <em>günstig</em>|10000|
```

Dateiformatsprobleme

- Nachteile
 - Bei beiden: Unterscheidung von Feldtrenner # von Nichtfeldtrennern nötig (z.B. reg. Ausdrücke für alle Spezialfälle)
1. Fehlerhafte Rückübersetzungen (color= kommt als Text irgendwo vor)
 2. Funktioniert nur falls | wirklich nirgends vorkommen kann

Manfred Mustermann#Villa am Strand#sehr schöne
Aussicht, günstig#10000#

1. a) &#n; in &n; ersetzen, color=# in color=

Manfred Mustermann#Villa am Strand#sehr schöne
Aussicht, günstig#10000#

b) &#n; in &#n; color= in color=# zurückübersetzen

sehr schöne
Aussicht, günstig

2. Alle Feldtrenner # durch z.B. | ersetzen

Manfred Mustermann|Villa am Strand|sehr schöne
Aussicht, günstig|10000|

Inhalt

- Integration Immobilienanwendung
- Proprietäres Textdateiformat für FTP Schnittstelle
 - Agregierte Information
 - Zuordnung der Daten
 - Umlaute
 - Falsch zugeordnete Werte
 - Dateiformatsprobleme
- Fazit

Integration Immobilienanwendung

- Fazit bei Anbindung von Fremdsystemen mit fixem Format
 - Nichts beim Vertrag an Anforderungen vergessen
 - Vollständige Spezifikation existierende Schnittstellen *vor* Vertragsabschluss fordern und begutachten
 - Abgrenzungen und Mitwirkungspflichten des Kunden müssen in den Vertrag
 - Kunde muss alle für die Entwicklung relevanten Informationen für die Schnittstellen auf Anfragen innerhalb 3 Arbeitstage liefert, sonst fallen Extrakosten an
 - Schnittstellen dürfen sich während der Entwicklung nicht ändern (freeze), sonst ...
 - Testsysteme für angebundene Systeme nötig für Entwicklung
- **Abbildung von Quell- und Zielfeldern definieren (Spezifikation) und vom Kunden bestätigen / abnehmen lassen**
- Bei Entwicklung eigener Schnittstellen
 - Standardformate verwenden
 - z.B. XML + W3C Schemata