

# 5. Mentorium

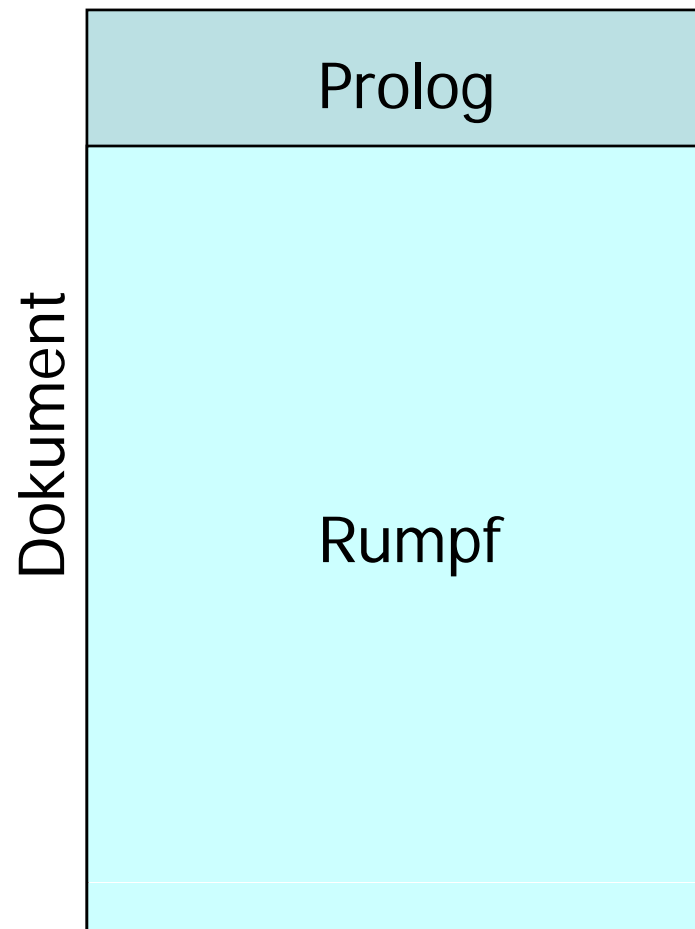
## XML, ERM & Normalisierung

Lösungshinweise

- Beschreiben Sie kurz die Eigenschaften eines XML-Dokuments. Grenzen Sie XML dabei zusätzlich von HTML ab.
- Welche Funktion hat eine DTD?
- Was ist ein "wohlgeformtes" bzw. "valides" XML Dokument?
- Welche Funktion hat ein Parser?

## Merkmale von XML

- Einfache und menschenlesbare Syntax (nicht binär)
- Standardisiert
- Selbstbeschreibend durch enthaltene Meta-Beschreibung
- Erweiterbar durch neue Elementbeschreibungen → anwendungsspezifische Datenmodelle
- Eignet sich zur Datenhaltung



Prolog enthält XML-Version und Angaben zum verwendeten Zeichensatz.

Rumpf enthält die Daten.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
```

Prolog

```
<flirt>  
  <name>Daisy</name>  
  <mobilfon>+436508469249</mobilfon>  
  <email>daisy@m-lehrstuhl.de</email>  
  <stadt>Innsbruck</stadt>  
  <erstkontakt>20.6.2006</erstkontakt>  
  <letztkontakt>14.7.2007</letztkontakt>  
  <geburtstag>12.7.1978</geburtstag>  
  <vegetarier>nein</vegetarier>  
  <gebunden>nein</gebunden>  
</flirt>
```

Rumpf

- HTML ist eine sehr einfache Beschreibungssprache:
  - Kaum semantische Beschreibung der Inhalte (nur Unterteilung in Abschnitte, Überschriften, Listen, etc.)
  - Dies macht automatische Verarbeitung von Webinhalten oft unmöglich.
- Z.B. woran erkennt man automatisch die Postadresse auf einer Webseite?

<H1>Lehrstuhl für M-Business und Mehrseitige Sicherheit</H1>

<P>Grüneburgplatz 1</P>

<H2>60629 Frankfurt </H2>

- Beim Erzeugen von HTML-Seiten aus Datenbanken ist erheblicher Aufwand nötig (Skripte, Formatierung, Darstellung).
- Dadurch redundante Datenhaltung, Gefahr von Inkonsistenzen
- HTML eignet sich nicht zur Datenhaltung, da die Sprache sich nicht eignet, Daten zu beschreiben.
- Bei HTML hat sich unsaubere Syntax etabliert – Elemente werden z.B. oft nicht abgeschlossen
- Es besteht also Bedarf an einer Beschreibung von Daten, welche diese Mängel beseitigt.

- Beschreiben Sie kurz die Eigenschaften eines XML-Dokuments. Grenzen Sie XML dabei zusätzlich von HTML ab.

- Welche Funktion hat eine DTD?

- Was ist ein "wohlgeformtes" bzw. "valides" XML Dokument?

- Welche Funktion hat ein Parser?

- **DTD**  
Document Type Definition – beschreibt die formale Struktur eines Dokuments
- **XML Schema**  
Alternativer Ansatz mit Erweiterungen der DTD
- **Parser**  
Übersetzt XML-Dokument in einen Dokumentenbaum
- **Style Sheet**  
Layout-Vorgaben zur Ausgabe von Dokumenten
- **Style-Sheet-Prozessor**  
Setzt die Style-Vorgaben um und generiert die Ausgabeseiten

- Eine Document Type Definition (DTD) beschreibt Struktur und Grammatik von XML-Dokumenten.
- Ist vergleichbar mit einer Variablen- / Typendeklaration in einer Programmiersprache
- Es wird definiert, welche Werte in Elementen vorkommen dürfen, damit ein „**gültiges**“ **XML-Dokument** entsteht.
- DTD „übersetzt“ also ein XML-Dokument in Datentypen für Datenbanken und erzeugt Regeln für Elemente.

- Beschreiben Sie kurz die Eigenschaften eines XML-Dokuments. Grenzen Sie XML dabei zusätzlich von HTML ab.
- Welche Funktion hat eine DTD?
- Was ist ein "wohlgeformtes" bzw. "valides" XML Dokument?
- Welche Funktion hat ein Parser?

- Wegen der eindeutigen, baumartigen Struktur und der Ähnlichkeit zu objektorientierten Systemen erkennt ein Rechner beim Einlesen von XML die Datenstruktur eindeutig.
- Ein „**wohlgeformtes Dokument**“ in XML ist ein syntaktisch korrektes Dokument mit abgeschlossenen, nicht überlappenden Tags und einheitlicher Groß- und Kleinschreibung.

- Beschreiben Sie kurz die Eigenschaften eines XML-Dokuments. Grenzen Sie XML dabei zusätzlich von HTML ab.
- Welche Funktion hat eine DTD?
- Was ist ein "wohlgeformtes" bzw. "valides" XML Dokument?
- Welche Funktion hat ein Parser?

- Zur Verarbeitung benötigt man einen Parser.
- Ein Parser ist eine Software, welche DTDs, Schemas und XML-Dokumente einlesen kann, um dann einer Anwendung Zugriffe auf alle Elemente zu ermöglichen.
- Übliches Vorgehen:
  - Anwendung öffnet XML-Datensatz.
  - Parser liest XML-Dokument und nötige DTDs, Schemas.
  - Parser bietet Anwendung Schnittstellen mit Funktionen wie "ElementeAuflisten()".
  - Anwendung durchsucht mittels der Schnittstellen das Dokument und bearbeitet die Elemente.
  - Anwendung speichert den überarbeiteten XML-Datensatz ab.
- Ergebnis  
Ein syntaktisch fehlerfreies und gegen eine DTD geprüftes XML-Dokument wird als "gültig" bzw. "valid" bezeichnet.

- Erstellen Sie für die Speicherung eines dynamischen Kundenprofils eine DTD und daraufbasierend ein XML Beispieldokument. Es sollen die persönlichen Daten eines Nutzers erfasst werden. Die persönlichen Daten sind dabei nochmal unterteilt in *Pseudonym*, *Alter*, *Geschlecht* und eine kommaseparierte Liste der *Interessen* des Nutzers.
- **Beispiel:**
  - Pseudonym: mobilerFritz1380, Alter: 25, Geschlecht: männlich
  - Interessen: Kino, Restaurants, Tennis, ...

- Element-Content:

EMPTY	leeres Element
ANY	beliebiger Inhalt
	Auswahlliste
,	Sequenz
()	Gruppierung
(#PCDATA)	Zeichen- oder Stringdaten

- Kardinalität:

	leer: genau ein Wert nötig
+	mindestens ein Wert
?	Null oder ein Wert
*	Null oder mehr Werte

## DTD:

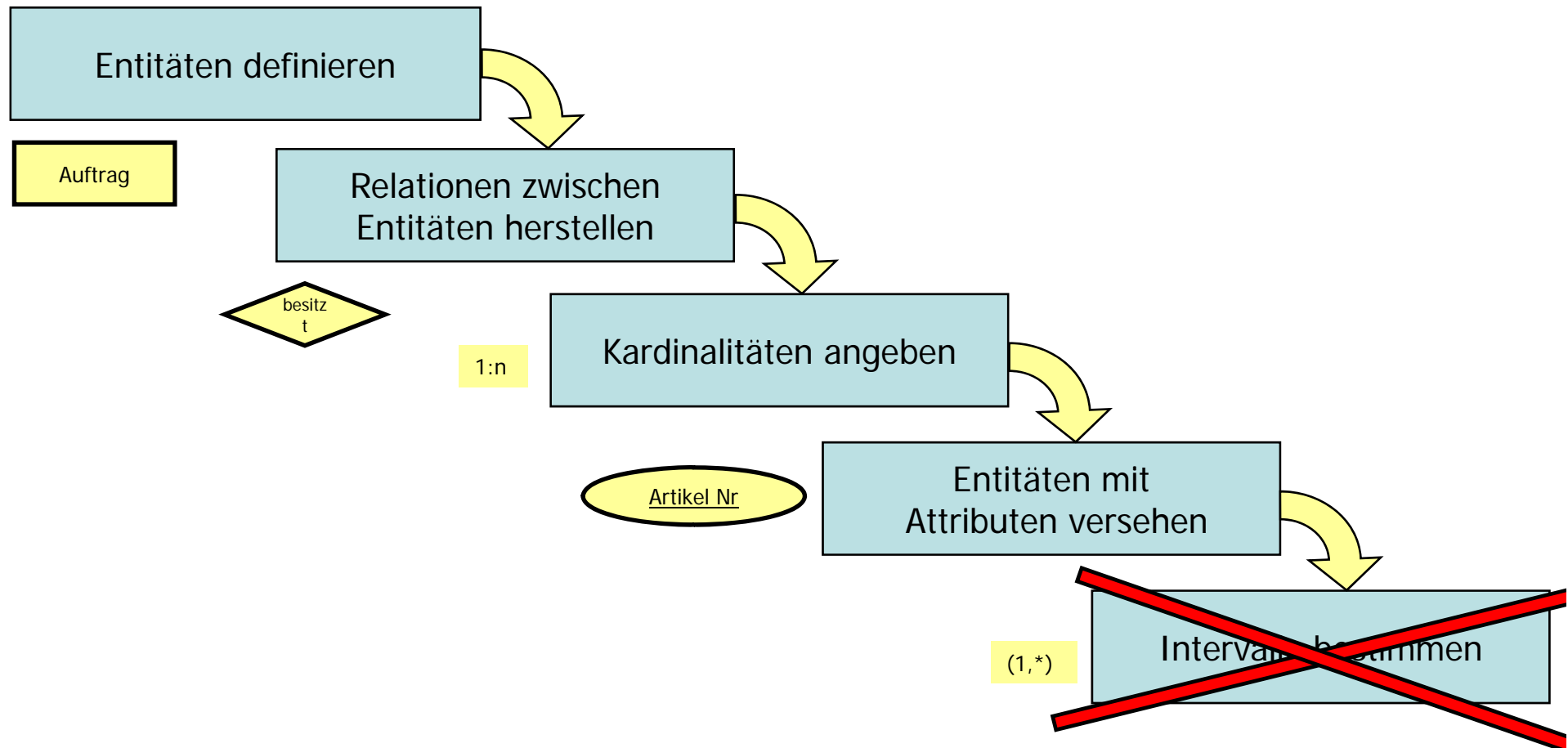
```
<!ELEMENT Benutzerprofil (pseudonym, geschlecht, interessen) >  
<!ELEMENT pseudonym (#PCDATA) >  
<!ELEMENT geschlecht(männlich | weiblich) >  
<!ELEMENT interessen (kino | restaurants | tennis | fußball |  
schwimmen | hockey)* >
```

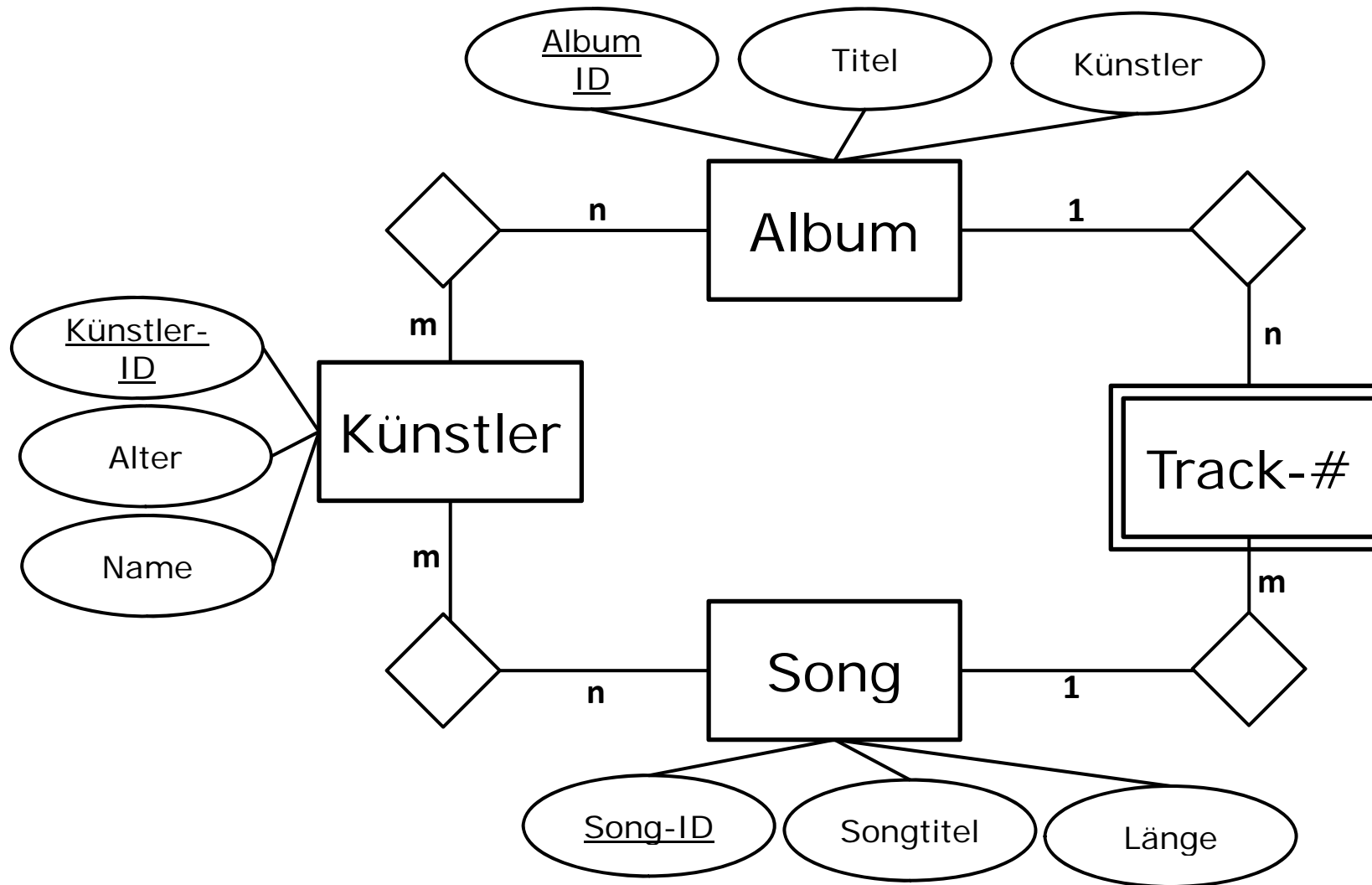
## XML-Dokument:

```
<?xml version="1.0" encoding = "ISO-8859-1" ?>  
<Benutzerprofil>  
  <pseudonym> mobiler fritz 1380 </pseudonym>  
  <alter> 25 </alter>  
  <interessen> kino, restaurants, tennis </interessen>  
</Benutzerprofil>
```

- Sie wollen Ihre private CD Sammlung künftig mit Hilfe einer Datenbank verwalten. Ein befreundeter Informatikstudent bietet Ihnen an dies für Sie zu erledigen, wenn Sie ihm anhand eines ERM den prinzipiellen Aufbau erläutern. Spontan fallen ihnen folgende Datentypen ein, die sie berücksichtigen wollen: Künstler, Album, Song, Track-#, Erscheinungsjahr.
- Geben Sie eine datenorientierte Sichtweise auf Ihre CD Sammlung. Erstellen Sie Relationstypen, Kardinalitäten und versehen Sie die Entitätstypen mit (Schlüssel-)Attributen.

# Vorgehen zur Erstellung eines ER-Modells (Bsp.)





- Bei einem Besuch schauen Sie sich die Musikdatenbank eines Freundes an. Gleichwohl sie von den Inhalten begeistert sind, sind sie es von der Struktur der Datenbank nicht.

CD-ID	Album	Titelliste
001	Michael Jackson – Dangerous (1991)	{1. Jam, 8. Black or White, 9. Gone to soon}
002	Queen – A Kind of Magic (1986)	{1. One Vision (Extended Version), 9. Princess of the Universe}
003	U2 – No Line on the Horizon (2009)	{2. Magnificent, 6. Get on your Boots}

- Mit welchen Argumenten können Sie Ihren Freund überzeugen, dass eine Normalisierung der Relation sinnvoll ist?
- Überführen Sie die Relation "CD\_Sammlung" in die 1., 2. und 3. Normalform

- Redundanz
  - Z.B. Mehrfacheinträge für die Zuordnung des Buches „Wirtschaftsinformatik“ zu der Signatur „QH1“.
- Einfügeanomalie
  - Z.B. das Sachgebiet kann nur aufgenommen werden, wenn ein Buch hierfür vorliegt.
- Änderungsanomalie
  - Z.B. Änderungen am Titel eines Buches sind mehrfach durchzuführen.
- Löschanomalie
  - Z.B. mit dem Löschen des letzten Buches wird auch das Sachgebiet selbst gelöscht.

- Bei einem Besuch schauen Sie sich die Musikdatenbank eines Freundes an. Gleichwohl sie von den Inhalten begeistert sind, sind sie es von der Struktur der Datenbank nicht.

CD-ID	Album	Titelliste
001	Michael Jackson – Dangerous (1991)	{1. Jam, 8. Black or White, 9. Gone to soon}
002	Queen – A Kind of Magic (1986)	{1. One Vision (Extended Version), 9. Princess of the Universe}
003	U2 – No Line on the Horizon (2009)	{2. Magnificent, 6. Get on your Boots}

- Mit welchen Argumenten können Sie Ihren Freund überzeugen, dass eine Normalisierung der Relation sinnvoll ist?
- Überführen Sie die Relation "CD\_Sammlung" in die 1., 2. und 3. Normalform

## 0. → 1. Normalform

Eine Relation ist dann in der 1. Normalform, wenn die Wertebereiche aller Attribute elementar sind.

Nichtelementare Wertebereiche der Attribute Album, Titelliste

Aufspalten der Attribute mit nichtelementarem Wertebereich in Attribute mit elementarem Wertebereich.

- Album → Albumtitel, Künstler, Erscheinungsjahr
- Titelliste → Track-#, Titel

## 1. Normalform:

### CD\_SAMMLUNG

CD-ID	Albumtitel	Künstler	Erscheinungsjahr	Track-#	Titel
001	Dangerous	Jackson, Michael	1991	1	Jam
001	Dangerous	Jackson, Michael	1991	8	Black or White
001	Dangerous	Jackson, Michael	1991	9	Gone too soon
002	A Kind of Magic	Queen	1986	1	One Vision (Extended Version)
002	A Kind of Magic	Queen	1986	9	Princess of the Universe
003	No Line on the Horizon	U2	2009	2	Magnificent
003	No Line on the Horizon	U2	2009	6	Get on Your Boots

## 1. → 2. Normalform

Eine Relation ist in der 2. Normalform, wenn sie in der 1. Normalform ist und jedes Nichtschlüsselattribut von dem Schlüsselkandidaten voll funktional abhängig ist.

- Exkurs: Grundlagen der Normalisierung

Primärschlüssel der Relation ist aus CD-ID und Track-# zusammengesetzt

Nichtschlüsselattribute nicht von komplettem Schlüsselkandidat komplett funktional abhängig.

- Aufsplitten der Relation

- 2. Normalform

**ALBUM**

CD-ID	Albumtitel	Künstler	Erscheinungsjahr
001	Dangerous	Jackson, Michael	1991
001	Dangerous	Jackson, Michael	1991
001	Dangerous	Jackson, Michael	1991
002	A Kind of Magic	Queen	1986
002	A Kind of Magic	Queen	1986
003	No Line on the Horizon	U2	2009
003	No Line on the Horizon	U2	2009

**TITEL\_LISTE**

Track-#	Titel
1	Jam
8	Black or White
9	Gone too soon
1	One Vision (Extended Version)
9	Princess of the Universe
2	Magnificent
6	Get on Your Boots

- 2. Normalform:

### ALBUM

CD-ID	Albumtitel	Künstler	Erscheinungsjahr
001	Dangerous	Jackson, Michael	1991
002	A Kind of Magic	Queen	1986
003	No Line on the Horizon	U2	2009

### TITEL\_LISTE

CD-ID	Track-#	Titel
001	1	Jam
001	8	Black or White
001	9	Gone too soon
002	1	One Vision (Extended Version)
002	9	Princess of the Universe
003	2	Magnificent
003	6	Get on Your Boots

## 2. → 3. Normalform

Eine Relation ist in der 3. Normalform, wenn sie in der 2. Normalform ist und kein Nichtschlüsselattribut transitiv von einem Schlüsselkandidaten abhängig ist.

Lat. "transitus" = Übergang

$A \rightarrow B, B \rightarrow C$ , dann ist C transitiv von A abhängig ( $A \rightarrow C$ )

CD-ID → Albumtitel, Albumtitel → Künstler & Erscheinungsjahr, folglich sind Erscheinungsjahr & Künstler transitiv von CD-ID abhängig (CD-ID → Erscheinungsjahr) und die o.g. Bedingung ("kein Nichtschlüsselattribut transitiv von einem Schlüsselkandidaten abhängig") verletzt).

- 3. Normalform

**ALBUM\_ZUORDNUNG**

CD-ID	Albumtitel
001	Dangerous
002	A Kind of Magic
003	No Line on the Horizon

**ALBUM**

Albumtitel	Künstler	Erscheinungsjahr
Dangerous	Jackson, Michael	1991
A Kind of Magic	Queen	1986
No Line on the Horizon	U2	2009

**TITEL\_LISTE**

CD-ID	Track-#	Titel
001	1	Jam
001	8	Black or White
001	9	Gone too soon
002	1	One Vision (Extended Version)
002	9	Princess of the Universe
003	2	Magnificent
003	6	Get on Your Boots

# Offene Fragen?