

## Problematik

Bei Anwendung des bisherigen Verfahrens zur Durchführung von Datenstrukturänderungen müssen Datenbanken, die bereits Anwenderdaten enthalten, zunächst durch individuell erstellte Selektionsprogramme entladen werden. Anschließend müssen die entladenen Datenbestände hinsichtlich der neuen Strukturen konvertiert werden. Vor der Generierung der neuen Datenstrukturen durch Löschen (DROP) und erneutes Erzeugen (CREATE) müssen alle Anwendungen, die mit dieser Datenbank verbunden sind, beendet werden. Nach erfolgreicher Neugenerierung müssen die entladenen Datenbestände durch individuelle Ladeprogramme wieder in die Datenbank eingefügt werden.

Diese Vorgehensweise erfordert einerseits einen hohen Einsatz von Entwicklungskapazitäten und verursacht andererseits häufig unverhältnismäßig lange Ausfallzeiten bezüglich der eingesetzten Datenbankanwendungen.

Solche Aktionen können betriebsbedingt häufig nur nachts oder an Wochenenden durchgeführt werden.

Das Erweitern bzw. Erzeugen von Tabellenstrukturen durch Löschen der vorhandenen Elemente (DROP) und ein sich anschließendes erneutes bzw. erstmaliges Erzeugen (CREATE) der veränderten Strukturen ist deshalb regelmäßig nur während der Entwicklungsphase einer gänzlich neuen Datenbanklösung möglich bzw. sinnvoll.

## Zusätzliche Möglichkeiten

1. Neue Spalten können an das Ende einer vorhandenen Tabellendefinition angefügt werden.
2. Vorhandene Tabellenstrukturen können durch neue Fremdschlüsseldefinitionen erweitert werden.

## Vorgehensweise beim Einfügen neuer Spalten

1. Anhängen der neuen Spalten an die vorhandenen Spalten einer bestimmten Tabelle

### Oracle

```
ALTER TABLE tblname ADD colname datatype[(maxlen[,decimals])];
```

### Interbase

```
ALTER TABLE tblname ADD colname datatype[(maxlen[,decimals])] [NOT NULL];
```

### Anmerkungen:

Hinzugefügte Felder werden durch Oracle in allen vorhandenen Datensätzen der Tabelle mit NULL initialisiert. Deshalb ist die Angabe der Zusatzklausel NOT NULL bei dieser Aktion in ORACLE nicht möglich.

Interbase belegt neue Felder vom Typ VARCHAR in vorhandenen Datensätzen mit einem leeren String. Numerische Felder werden mit 0, Felder für Datum/Zeit-Werte mit einem Pseudodatum 17.11.1858 belegt.

2. Versorgen der neuen Felder mit selbstgewählten Standardwerten bzw. NULL

```
UPDATE tblname SET colname=coldata[,colname=coldata,...] [WHERE...];
```

### Anmerkungen zu Interbase:

Die hinzugefügten Felder sind zwar mit Werten ungleich NULL belegt, müssen aber in Abfragebedingungen wie NULL-Konstanten behandelt werden. Erst nach der ersten mit COMMIT abgeschlossenen UPDATE-Anweisung können Bedingungen zu den Daten hinzugefügter Felder mit den bekannten Vergleichsoperatoren formuliert werden.

D.h.: Interbase benötigt in jedem Fall eine entsprechende Update-Anweisung mit anschließendem COMMIT.

### 3. Erzeugen der endgültigen Spaltendefinitionen, falls nicht bereits gegeben

Oracle (nur falls Klausel NOT NULL bzw. Check-Bedingung erforderlich)

```
ALTER TABLE tblname MODIFY colname datatype[(maxlen[,decimals])] [NOT NULL] [CHECK (...)];
```

Interbase (nur falls Check-Bedingung erforderlich)

```
ALTER TABLE tblname ADD CHECK (...)
```

### 4. Erzeugen der notwendigen Fremdschlüsselbeziehungen

```
ALTER TABLE tblname ADD FOREIGN KEY(colname[,colname,...]) REFERENCES tblname  
[ON UPDATE {SET NULL|CASCADE}] [ON DELETE {SET NULL|CASCADE}];
```

Anmerkung: Die Klausel ON UPDATE ... ist in Oracle nicht verfügbar.

## Beispiel

Die Datenbank soll um eine Tabelle für Umsatzsteuerschlüssel (UST) und um eine zweite Tabelle für Branchenschlüssel (BRA) erweitert werden. An die Lieferantentabelle (LIE) soll eine Spalte für den neuen Umsatzsteuerschlüssel, die nicht NULL sein darf, und eine Spalte für den Branchenschlüssel, die auch NULL beinhalten kann, angehängt werden.

Der Umsatzsteuerschlüssel der Lieferanten soll, abhängig vom Länderkennzeichen, automatisch belegt werden:

- allen Inlandslieferanten (LAND IS NULL) soll der Umsatzsteuerschlüssel 'Inland' zugewiesen werden.
- allen Lieferanten aus Ländern der EU (LAND IN ('DK','E','F','GB','I','NL')) soll der Umsatzsteuerschlüssel 'EU' zugewiesen werden.
- allen anderen Lieferanten soll der Umsatzsteuerschlüssel 'Ausland' zugewiesen werden.

Das Entfernen eines Eintrags aus der Umsatzsteuerschlüsseltabelle darf nicht möglich sein, solange noch entsprechende Verweise in der Lieferantentabelle enthalten sind.

Die neue Spalte Branchenschlüssel in der Lieferantentabelle soll für alle Lieferanten mit NULL initialisiert werden. Beim Löschen eines Branchenschlüssels aus der Branchenschlüsseltabelle sollen alle diesbezüglichen Verweise in der Lieferantentabelle automatisch mit NULL belegt werden.

### 1. Anlegen der Schlüsseltabellen

Oracle

```
CREATE TABLE UST (USTSCHL VARCHAR2(10) NOT NULL,  
PRIMARY KEY (USTSCHL));
```

```
CREATE TABLE BRA (BRANCHE VARCHAR2(20) NOT NULL,  
PRIMARY KEY (BRANCHE));
```

Interbase

```
CREATE TABLE UST (USTSCHL VARCHAR(10) NOT NULL CHECK(USTSCHL<>''),  
PRIMARY KEY (USTSCHL));
```

```
CREATE TABLE BRA (BRANCHE VARCHAR(20) NOT NULL CHECK(BRANCHE<>''),  
PRIMARY KEY (BRANCHE));
```

### 2. Einfügen der für die Konvertierung der Lieferantentabelle notwendigen Umsatzsteuerschlüssel

```
INSERT INTO UST VALUES('Inland');  
INSERT INTO UST VALUES('Ausland');  
INSERT INTO UST VALUES('EU');
```

### 3. Anhängen der neuen Spalten an die Lieferantentabelle

#### Oracle

```
ALTER TABLE LIE ADD USTSCHL VARCHAR2(10);  
ALTER TABLE LIE ADD BRANCHE VARCHAR2(20);
```

#### Interbase

```
ALTER TABLE LIE ADD USTSCHL VARCHAR(10) NOT NULL;  
ALTER TABLE LIE ADD BRANCHE VARCHAR(20);
```

### 4. Versorgen der neuen Felder mit selbstgewählten Standardwerten bzw. NULL

```
UPDATE LIE SET USTSCHL='Inland' WHERE LAND IS NULL;  
UPDATE LIE SET USTSCHL='EU' WHERE LAND IN ('DK','E','F','GB','I','NL');  
UPDATE LIE SET USTSCHL='Ausland' WHERE USTSCHL IS NULL;
```

#### Zusätzlich in Interbase:

```
UPDATE LIE SET BRANCHE=NULL;
```

### 5. Datensatzänderungen per Anweisung COMMIT permanent in der Datenbank speichern

```
COMMIT;
```

### 6. Erzeugen der endgültigen Spaltendefinitionen

#### Oracle

```
ALTER TABLE LIE MODIFY USTSCHL VARCHAR2(10) NOT NULL;
```

#### Interbase

```
ALTER TABLE LIE ADD CHECK(USTSCHL<>'');  
ALTER TABLE LIE ADD CHECK(BRANCHE IS NULL OR BRANCHE<>'');
```

### 7. Erzeugen der notwendigen Fremdschlüsselbeziehungen

```
ALTER TABLE LIE ADD FOREIGN KEY (USTSCHL) REFERENCES UST;  
ALTER TABLE LIE ADD FOREIGN KEY (BRANCHE) REFERENCES BRA ON DELETE SET NULL;
```

## Löschen von Spalten- bzw. Fremdschlüsseldefinitionen

Fremdschlüssel können mit nachfolgender Anweisung entfernt werden:

```
ALTER TABLE tblname DROP CONSTRAINT keyname;
```

Die Ermittlung des Fremdschlüsselnamens (keyname) ist nur über die Systemtabellen des Datenbanksystems möglich und mit erheblichen Schwierigkeiten behaftet, da die Struktur der Systemtabellen häufig sehr komplex und leider nicht normiert ist.

Spalten können, nachdem eventuelle Fremdschlüsselverweise eliminiert wurden, mit nachfolgender Anweisung entfernt werden:

```
ALTER TABLE tblname DROP COLUMN colname;
```

Tip: Änderungen an den Datenbankstrukturen sollten mit absoluter Sorgfalt geplant und vorbereitet werden, um komplizierte Fehlerkorrekturen mit Befehlen des Typs **ALTER TABLE tblname DROP ...** zu vermeiden.

## Übungsaufgabe

Die Lieferantentabelle (LIE) enthält z.Zt. keinerlei Adressdaten.

Die folgenden Spalten sollen deshalb an die Lieferantentabelle angefügt werden:

Name	Datentyp	Länge	Kommentar
NAME1	Beliebige Zeichen	40	Erster Teil des Firmennamens (Null nicht zulässig)
NAME2	Beliebige Zeichen	40	Zweiter Teil des Firmennamens (Null zulässig)
STRASSE	Beliebige Zeichen	40	Straße und Hausnummer (Null nicht zulässig)
PLZORT	Beliebige Zeichen	40	Postleitzahl und Ort (Null nicht zulässig)

Die Spalten NAME1, STRASSE und PLZORT sollen in den vorhandenen Datensätzen mit dem Zeichen '\*' als Indikator für manuellen Änderungsbedarf belegt werden. Die Spalte NAME2 soll mit NULL initialisiert werden.

Das Entladen, Generieren und erneute Laden der Datenbank ist wegen der damit verbundenen langen Ausfallzeit nicht möglich.

## Lösung

### Oracle

```
ALTER TABLE LIE ADD NAME1 VARCHAR2(40);
ALTER TABLE LIE ADD NAME2 VARCHAR2(40);
ALTER TABLE LIE ADD STRASSE VARCHAR2(40);
ALTER TABLE LIE ADD PLZORT VARCHAR2(40);

UPDATE LIE SET NAME1='*', STRASSE='*', PLZORT='*';
COMMIT;

ALTER TABLE LIE MODIFY NAME1 VARCHAR2(40) NOT NULL;
ALTER TABLE LIE MODIFY STRASSE VARCHAR2(40) NOT NULL;
ALTER TABLE LIE MODIFY PLZORT VARCHAR2(40) NOT NULL;
```

### Interbase

```
ALTER TABLE LIE ADD NAME1 VARCHAR(40) NOT NULL;
ALTER TABLE LIE ADD NAME2 VARCHAR(40);
ALTER TABLE LIE ADD STRASSE VARCHAR(40) NOT NULL;
ALTER TABLE LIE ADD PLZORT VARCHAR(40) NOT NULL;

UPDATE LIE SET NAME1='*', NAME2=NULL, STRASSE='*', PLZORT='*';
COMMIT;

ALTER TABLE LIE ADD CHECK(NAME1<>'');
ALTER TABLE LIE ADD CHECK(NAME2 IS NULL OR NAME2<>'');
ALTER TABLE LIE ADD CHECK(STRASSE<>'');
ALTER TABLE LIE ADD CHECK(PLZORT<>'');
```

## Übungsaufgabe

Bestellungen bei ausländischen Lieferanten verursachen häufig, aufgrund der unterschiedlichen Sprachen, einen erhöhten nachträglichen Klärungsaufwand. Aus diesem Grund sollen die Artikelbezeichnungen (ART.BEZEICHN) in den gängigen Fremdsprachen hinterlegt werden. Außerdem soll ausländischen Lieferanten eine Fremdsprache zugeordnet werden können. Diese Maßnahme soll in einer späteren Ausbaustufe das Drucken fremdsprachlicher Bestellungen ermöglichen.

### Teilaufgabe a

Die gängigen Fremdsprachen sollen in einer Schlüsselstabelle SPR gespeichert werden:

Name	Datentyp	Länge	Kommentar
SPRACHE	Beliebige Zeichen	10	Primärschlüssel

Die Lieferantentabelle (LIE) soll um eine Spalte SPRACHE, die auch NULL beinhalten darf, ergänzt werden. Das Löschen bzw. Ändern einer Sprache in der Schlüsselstabelle darf nicht möglich sein, solange noch entsprechende Verweise in der Lieferantentabelle gespeichert sind.

Abhängig vom Länderkennzeichen soll den vorhandenen Lieferanten erstmalig eine Sprache zugeordnet werden:

- allen Lieferanten mit Länderkennzeichen 'BM', 'GB' oder 'USA' soll die Sprache 'Englisch' zugeordnet werden.
- allen Lieferanten mit Länderkennzeichen 'F' soll die Sprache 'Franzoes' zugeordnet werden.
- bei allen anderen Lieferanten soll das Feld Sprache mit NULL initialisiert werden.

Das Entladen, Generieren und erneute Laden der Datenbank ist wegen der damit verbundenen langen Ausfallzeit nicht möglich.

## Lösung

### Oracle

```
CREATE TABLE SPR (SPRACHE VARCHAR2(10) NOT NULL,
                  PRIMARY KEY (SPRACHE));

INSERT INTO SPR VALUES('Englisch');
INSERT INTO SPR VALUES('Franzoes');

ALTER TABLE LIE ADD SPRACHE VARCHAR2(10);

UPDATE LIE SET SPRACHE='Englisch' WHERE LAND IN ('BM', 'GB', 'USA');
UPDATE LIE SET SPRACHE='Franzoes' WHERE LAND='F';
COMMIT;

ALTER TABLE LIE ADD FOREIGN KEY (SPRACHE) REFERENCES SPR;
```

### Interbase

```
CREATE TABLE SPR (SPRACHE VARCHAR(10) NOT NULL CHECK(SPRACHE<>''),
                  PRIMARY KEY (SPRACHE));

INSERT INTO SPR VALUES('Englisch');
INSERT INTO SPR VALUES('Franzoes');

ALTER TABLE LIE ADD SPRACHE VARCHAR(10);

UPDATE LIE SET SPRACHE=NULL;
UPDATE LIE SET SPRACHE='Englisch' WHERE LAND IN ('BM', 'GB', 'USA');
UPDATE LIE SET SPRACHE='Franzoes' WHERE LAND='F';
COMMIT;

ALTER TABLE LIE ADD CHECK(SPRACHE IS NULL OR SPRACHE<>'');
ALTER TABLE LIE ADD FOREIGN KEY (SPRACHE) REFERENCES SPR;
```

## Übungsaufgabe

### Teilaufgabe b

- Entwerfen Sie eine Tabelle AFS zur Speicherung der fremdsprachlichen Artikelbezeichnungen. Beim Löschen eines Artikels sollen die fremdsprachlichen Bezeichnungen automatisch entfernt werden. Das Löschen einer Sprache aus der Schlüsseltabelle darf nicht möglich sein, solange noch entsprechende Verweise in der Tabelle der fremdsprachlichen Bezeichnungen gespeichert sind. Fügen Sie mindestens 4 Beispielsätze in die neue Tabelle ein.
- Erzeugen Sie eine Sql-Abfrage zur Selektion aller Artikel mit fremdsprachlichen Bezeichnungen. Die Ausgabe soll die Artikelnummer, die deutschsprachige Bezeichnung mit der Spaltenüberschrift 'DEUTSCH', die Sprache und die fremdsprachliche Bezeichnung enthalten sowie nach Artikelnummer und Sprache sortiert sein.

Ausgabebeispiel:

ARTIKEL	DEUTSCH	SPRACHE	BEZEICHN
10.1001	Gartenpumpe	Englisch	garden-pump
10.1001	Gartenpumpe	Franzoes	Pompent de jardin
10.1016	Hochdruckreiniger	Englisch	high-pressure-cleaner
10.1016	Hochdruckreiniger	Franzoes	Nettoyeuse Forte pression

## Lösung

### Aufgabe 1 (Oracle)

```
CREATE TABLE AFS (ARTIKEL VARCHAR2(10) NOT NULL,  
                  SPRACHE VARCHAR2(10) NOT NULL,  
                  BEZEICHN VARCHAR2(40) NOT NULL,  
                  FOREIGN KEY (ARTIKEL) REFERENCES ART ON DELETE CASCADE,  
                  FOREIGN KEY (SPRACHE) REFERENCES SPR,  
                  PRIMARY KEY (ARTIKEL,SPRACHE));  
  
INSERT INTO AFS VALUES('10.1001','Englisch','garden-pump');  
INSERT INTO AFS VALUES('10.1001','Franzoes','Pompent de jardin');  
INSERT INTO AFS VALUES('10.1016','Englisch','high-pressure-cleaner');  
INSERT INTO AFS VALUES('10.1016','Franzoes','Nettoyeuse Forte pression');
```

### Aufgabe 1 (Interbase)

```
CREATE TABLE AFS (ARTIKEL VARCHAR(10) NOT NULL,  
                  SPRACHE VARCHAR(10) NOT NULL,  
                  BEZEICHN VARCHAR(40) NOT NULL CHECK(BEZEICHN<>''),  
                  FOREIGN KEY (ARTIKEL) REFERENCES ART ON DELETE CASCADE,  
                  FOREIGN KEY (SPRACHE) REFERENCES SPR,  
                  PRIMARY KEY (ARTIKEL,SPRACHE));  
  
INSERT INTO AFS VALUES('10.1001','Englisch','garden-pump');  
INSERT INTO AFS VALUES('10.1001','Franzoes','Pompent de jardin');  
INSERT INTO AFS VALUES('10.1016','Englisch','high-pressure-cleaner');  
INSERT INTO AFS VALUES('10.1016','Franzoes','Nettoyeuse Forte pression');
```

### Aufgabe 2

```
SELECT AFS.ARTIKEL,ART.BEZEICHN AS DEUTSCH,AFS.SPRACHE,AFS.BEZEICHN FROM AFS,ART  
WHERE ART.ARTIKEL=AFS.ARTIKEL ORDER BY AFS.ARTIKEL,AFS.SPRACHE;
```