

Datenbank-Programmierung

Anhang B: ALTER TABLE in verschiedenen DBMS

Prof. Dr. Stefan Brass

Martin-Luther-Universität Halle-Wittenberg

Sommersemester 2018/19

<http://www.informatik.uni-halle.de/~brass/dbp19/>

Inhalt

- 1 Oracle
- 2 DB2
- 3 SQL Server
- 4 Access
- 5 MySQL

ALTER TABLE in Oracle (1)

Spalten hinzufügen:

- Z.B. Spalte "ZUSATZPKT" zu "STUDENTEN" hinzufügen:

```
ALTER TABLE STUDENTEN ADD  
(ZUSATZPKT NUMERIC(4,1) CHECK(ZUSATZPKT >= 0))
```

- Bei existierenden Zeilen ist die neue Spalte Null.

Natürlich kann man den Wert später verändern.

- Ist ein Default-Wert festgelegt, kann die neue Spalte NOT NULL sein:

```
ALTER TABLE STUDENTEN ADD  
(ZUSATZPKT NUMERIC(4,1) DEFAULT 0 NOT NULL  
CHECK(ZUSATZPKT >= 0))
```

ALTER TABLE in Oracle (2)

Spalten verändern:

- Der Datentyp einer Spalte kann verändert werden:

```
ALTER TABLE AUFGABEN MODIFY (THEMA VARCHAR(100))
```

- Die Breite einer Spalte kann nicht verringert werden, außer sie enthält nur Nullwerte.
- MODIFY kann nicht verwendet werden, um Spalten-Constraints zu ändern, außer NULL/NOT NULL.
Es gibt eine andere Syntax dafür, siehe nächste Folie.
- In Oracle können Spalten nicht gelöscht oder umbenannt werden.

ALTER TABLE in Oracle (3)

Constraints hinzufügen/entfernen:

- Einen Constraint hinzufügen:

```
ALTER TABLE STUDENTEN ADD  
    (CONSTRAINT SCH2 UNIQUE(VORNAME, NACHNAME))
```

- Hier muss die Syntax für Tabellen-Constraints verwendet werden.

Intern werden alle Constraints in Tabellen-Constraints übersetzt.

- Einen benannten Constraint entfernen:

```
ALTER TABLE STUDENTEN DROP CONSTRAINT SCH2
```

Man benötigt zum Entfernen meist den Namen des Constraints. Man kann ihn im Data Dictionary nachsehen (`USER_CONSTRAINTS`).

ALTER TABLE in Oracle (4)

Constraints entfernen, fortgesetzt:

- Primär- und Alternativschlüssel kann man entfernen, ohne den Namen zu kennen:

```
ALTER TABLE STUDENTEN DROP  
PRIMARY KEY CASCADE
```

```
ALTER TABLE STUDENTEN DROP  
UNIQUE(VORNAME, NACHNAME)
```

- NOT NULL-Constraints können mit einer Spaltenmodifikation entfernt werden:

```
ALTER TABLE AUFGABEN MODIFY (THEMA NULL)
```

- Man kann Constraints auch aktivieren/deaktivieren.

Eine deaktivierte Bedingung existiert noch im Data Dictionary, wird aber nicht überprüft/erzungen.

ALTER TABLE in Oracle (5)

Tabellen umbenennen:

- Tabellen können in Oracle umbenannt werden. Dies geschieht mit einem speziellen “RENAME”-Befehl:

```
RENAME STUDENTEN TO STUD
```

- Dies ist im SQL-92-Standard nicht enthalten.

Es funktioniert auch in DB2, aber nicht in SQL Server, MySQL oder Access. Oracle (aber nicht DB2) versteht auch

```
“ALTER TABLE STUDENTEN RENAME TO STUD”.
```

- Ich kenne keine Möglichkeit, Spalten oder DB-Nutzer in Oracle umzubenennen.

Wenn Sie es wissen, sagen Sie es mir.

Inhalt

- 1 Oracle
- 2 DB2
- 3 SQL Server
- 4 Access
- 5 MySQL

ALTER TABLE in DB2 (1)

Spalten hinzufügen:

- Das Hinzufügen von Spalten funktioniert wie im SQL-Standard:

```
ALTER TABLE STUDENTEN  
ADD COLUMN ZUSATZPKT NUMERIC(4,1)
```

- Ist ein Default-Wert festgelegt, kann die neue Spalte NOT NULL sein.
- Es gibt keine Möglichkeit, Spalten zu löschen oder umzubenennen.

ALTER TABLE in DB2 (2)

Spalten verändern:

- Die einzige erlaubte Modifikation einer Spalte ist die Änderung der Länge einer VARCHAR-Spalte:

```
ALTER TABLE AUFGABEN
```

```
ALTER THEMA SET DATA TYPE VARCHAR(100)
```

Es sind wieder nur Erhöhungen der Länge möglich.

- Es scheint so, dass der NULL/NOT NULL-Status nicht geändert werden kann.

Man kann NOT NULL mit einem CHECK-Constraint schreiben, aber das System versteht die Äquivalenz nicht richtig, z.B. verlangt PRIMARY KEY die Bedingung NOT NULL.

ALTER TABLE in DB2 (3)

Constraints hinzufügen/entfernen:

- Hinzufügen funktioniert wie im SQL-Standard:

```
ALTER TABLE STUDENTEN  
ADD CHECK(ZUSATZPKT IS NOT NULL)
```

- Ein benannter Constraint kann entfernt werden
(wie in SQL-92, aber ohne RESTRICT/CASCADE):

```
ALTER TABLE STUDENTEN  
DROP CONSTRAINT ZUSATZPKT_DEF
```

- Primärschlüssel kann man ohne Namen entfernen:

```
ALTER TABLE STUDENTEN DROP PRIMARY KEY
```

Inhalt

- 1 Oracle
- 2 DB2
- 3 SQL Server**
- 4 Access
- 5 MySQL

ALTER TABLE in SQL Server (1)

Spalten hinzufügen/löschen:

- Das Hinzufügen funktioniert wie im Standard, aber das Wort "COLUMN" nach "ADD" ist nicht erlaubt.

```
ALTER TABLE STUDENTEN ADD ZUSATZPKT NUMERIC(4,1)
```

Ist ein Default-Wert festgelegt, kann die neue Spalte NOT NULL sein.

- Eine Spalte kann wie folgt gelöscht werden (das Schlüsselwort "COLUMN" wird verlangt, RESTRICT oder CASCADE werden nicht verstanden):

```
ALTER TABLE STUDENTEN DROP COLUMN ZUSATZPKT
```

ALTER TABLE in SQL Server (2)

Spalten verändern:

- Der Datentyp einer Spalte kann verändert werden, aber nicht, wenn die Spalte ein Schlüssel ist oder ein Check-Constraint vorliegt:

```
ALTER TABLE AUFGABEN  
ALTER COLUMN THEMA VARCHAR(100)
```

Verringerungen der Größe sind möglich, wenn die Daten noch passen.

- Man kann auch die NULL/NOT NULL-Bedingung in Spalten ändern:

```
ALTER TABLE AUFGABEN  
ALTER COLUMN THEMA VARCHAR(100) NULL
```

ALTER TABLE in SQL Server (3)

Constraints hinzufügen/entfernen:

- Das Hinzufügen funktioniert wie im SQL-Standard:

```
ALTER TABLE STUDENTEN  
ADD CHECK(ZUSATZPKT IS NOT NULL)
```

Eine Spalte muss Not Null sein, um einen Primärschlüssel-Constraint hinzuzufügen.

- Eine benannte Bedingung kann entfernt werden:

```
ALTER TABLE STUDENTEN  
DROP CONSTRAINT ZUSATZPKT_DEF
```

(Wie in SQL-92, aber ohne RESTRICT/CASCADE).

Inhalt

- 1 Oracle
- 2 DB2
- 3 SQL Server
- 4 Access**
- 5 MySQL

ALTER TABLE in ACCESS

- Das Hinzufügen und Löschen von Constraints funktioniert wie im SQL-92-Standard.

Das Schlüsselwort `COLUMN` ist optional, auch wenn es im Handbuch anscheinend verlangt wird. `CASCADE` und `RESTRICT` werden im Handbuch beim Löschen von Constraints nicht erwähnt, aber sie werden akzeptiert (obwohl sie wie im Standard nicht unbedingt nötig sind).

- Der Datentyp einer Spalte kann vielseitig geändert werden. Man kann z.B. zwischen Zahlen und Zeichenketten konvertieren (wenn die Zeichenketten ein numerisches Format haben).

```
ALTER TABLE STUDENTEN ALTER COLUMN ZUSATZPKT CHAR(20)
```

Inhalt

- 1 Oracle
- 2 DB2
- 3 SQL Server
- 4 Access
- 5 MySQL**

ALTER TABLE in MySQL (1)

- MySQL hat ein sehr umfangreiches ALTER TABLE-Statement mit vielen Optionen (siehe Handbuch).

Dies liegt zum Teil daran, dass MySQL ALTER TABLE ausführt, indem eine Kopie der gesamten Tabelle mit der neuen Struktur erstellt wird. Das kann bei großen Tabellen aufwändig werden und andere Systeme versuchen, dies zu vermeiden (was zu vielen Einschränkungen führt).

- Die SQL-92-Syntax wird verstanden, außer beim Löschen von benannten Constraints.

Sogar RESTRICT/CASCADE werden zumindest syntaktisch für DROP COLUMN akzeptiert, auch wenn sie im Handbuch nicht aufgeführt sind.

ALTER TABLE in MySQL (2)

- Es gibt eine nicht standardisierte Syntax für das Löschen der Constraints, die MySQL unterstützt.

Mit `“ALTER TABLE T DROP PRIMARY KEY”` kann man den Primärschlüssel entfernen. Bei anderen Schlüsseln muss man den Namen *X* herausfinden, den MySQL dem Schlüssel bzw. Index zugeordnet hat (mit `“SHOW CREATE TABLE T”`), dann `“ALTER TABLE T DROP INDEX X”`. CHECK-Constraints und Fremdschlüssel werden nicht unterstützt. Der Datentyp einer Spalte und der NULL/NOT NULL-Status werden wie folgt geändert: `“ALTER TABLE AUFGABEN MODIFY THEMA VARCHAR(100) NULL”`.

- Tabellen und Spalten können umbenannt werden.

```
ALTER TABLE STUDENTEN RENAME TO STUD
ALTER TABLE STUDENTEN CHANGE SID STUD_NR
NUMERIC(3) PRIMARY KEY
```