

Referenzielle Integrität

in

SQL

aus „Referential Integrity Is Important For Databases“
von Michael Blaha (Modelsoft Consulting Corp)

Referenzielle Integrität

Definition:

Referenzielle Integrität bedeutet die Konsistenz zwischen verbundenen Tabellen. *Referenzielle Integrität* wird durch die Kombination von Primärschlüssel und Fremdschlüssel erzwungen. Um die *referenzielle Integrität* zu erhalten darf jedes Feld einer Tabelle, das als Fremdschlüssel deklariert worden ist, nur Werte des entsprechenden Primärschlüssels der „Eltern Tabelle“ annehmen.

Vorteile der Referenziellen Integrität

- **Steigerung der Daten-Qualität:** Referenzielle Integrität hilft Fehler zu vermeiden.
- **Schnellere Entwicklung:** Referenzielle Integrität muss nicht in jeder Applikation neu implementiert werden.
- **Weniger Fehler:** Einmal definierte referenzielle Integritätsbedingungen gelten für alle Applikation der selben Datenbank.
- **Konsistentere Applikationen:** Referenzielle Integrität ist für alle Applikation, die auf die selbe Datenbank zugreifen, gleich.

SQL Standard

ALTER TABLE table-name1

ADD CONSTRAINT constraint-name

FOREIGN KEY (column-list)

REFERENCES table-name2 [(column-list)]

[*DeleteAction*] [*UpdateAction*]

Delete Actions

- **Cascade:** Tupel, die Fremdschlüssel enthalten werden auch gelöscht.
- **No action:** Von Fremdschlüssel abhängige Tupel dürfen nicht gelöscht werden.
- **Set null:** Fremdschlüssel wird auf „*NULL*“ gesetzt.
- **Set default:** Fremdschlüssel wird auf einen Default-Wert gesetzt.

Anmerkung:

Die gleichen *Actions* können analog auch für *Update Actions* verwendet werden.

Beispiel

DELETE FROM Produkt
WHERE produktnr='1'

Cascade:

Kunde			Produkt	
<u>kundennr</u>	standort	produktnr	<u>produktnr</u>	beschreibung
1	Frankfurt	2	1	Telefon
2	München	1	2	FAX

No action:

Kunde			Produkt	
<u>kundennr</u>	standort	produktnr	<u>produktnr</u>	beschreibung
1	Frankfurt	2	1	Telefon
2	München	1	2	FAX

Set null:

Kunde			Produkt	
<u>kundennr</u>	standort	produktnr	<u>produktnr</u>	beschreibung
1	Frankfurt	2	1	Telefon
2	München	null	2	FAX

Set default:

Kunde			Produkt	
<u>kundennr</u>	standort	produktnr	<u>produktnr</u>	beschreibung
1	Frankfurt	2	1	Telefon
2	München	<defaultwert>	2	FAX

SQL Server

(Microsoft)

- Fremdschlüssel kann auf Primärschlüssel verweisen oder auf eine eindeutige Spalten-Kombination.
- Unterstützt folgende *Delete* und *Update Actions*:
 - *cascade*
 - *no action* (Syntax nicht SQL-Standard)
 - *set null* (ab Version 2005)
 - *set default* (ab Version 2005)
- „*no action*“ ist Default-Action.

Oracle (Version 10g)

- Fremdschlüssel kann auf Primärschlüssel verweisen oder auf eine eindeutige Spalten-Kombination.
- Bietet keine *Update Actions*.
- Unterstützt folgende *Delete Actions*:
 - *cascade*
 - *no action* (Syntax nicht SQL-Standard)
 - *set null*
- „*no action*“ ist Default-Action.

MySQL (Version 5)

- Fremdschlüssel kann auf Primärschlüssel verweisen oder auf eine eindeutige Spalten-Kombination.
- Fremdschlüssel und referenzierter Schlüssel müssen beide Datenbank Indizes besitzen.
- Unterstützt *Delete Actions* und *Update Actions*.
- Hält sich an den SQL-Standard.

MS-Access

- SQL-Engine unterstützt referenzielle Integrität nur teilweise.
- Mit Hilfe der grafischen Benutzeroberfläche können Actions (*cascade*, *no action*) definiert werden.