

# DEFAULT- und GENERATED-Werte

Das Dokument zeigt die Verhaltensweise von DEFAULT- und GENERATED-Werten auf. Insbesondere werden behandelt:

- **CREATE und ALTER TABLE**
  - WITH DEFAULT-Klausel und ihre Bedeutung
  - Besonderheiten beim Hinzufügen einer neuen Spalte mit DEFAULT-Werten.
  - GENERATED ALWAYS und GENERATED BY DEFAULT-Klausel und ihre Bedeutung
  - Unterstützte GENERATED-Typen:
    - IDENTITY
    - ROWID
    - ROW CHANGE TIMESTAMP
  - Besonderheiten beim Hinzufügen einer neuen Spalte mit GENERATED-Werten.
- **UNLOAD- und LOAD-Besonderheiten**
  - Behandlung von DEFAULT-Werten
  - Behandlung von GENERATED ALWAYS-Werten
  - Behandlung von GENERATED BY DEFAULT-Werten
- **Relevante Katalog-Informationen über DEFAULT- und GENERATED-Werte**
  - Katalog-Tabellen-Infos

Hinweise: Die Striche an der Seite zeigen die Veränderungen der DB2-Version 9 for z/OS auf.

© Copyright 2009 DGD mbH

Stand: 15.3.2009

DGD-Dienstleistungsgesellschaft für Datenverarbeitung mbH  
Im Weingarten 47

D-65201 Wiesbaden Frauenstein  
Tel. 0611 / 94 27 30  
Fax 0611 / 42 89 43  
Email [info@dgd-ub.de](mailto:info@dgd-ub.de)  
Homepage <http://www.dgd-ub.de>

Soweit nicht ausdrücklich von der DGD schriftlich zugestanden, verpflichtet eine Verwertung, Weitergabe, Vervielfältigung oder ein Nachdruck - auch auszugsweise - dieser Unterlage oder ihres Inhalts zu Schadenersatz (BGB, UWG, LitUrhG).



# Inhaltsverzeichnis

<b>1</b>	<b>DEFAULT- und GENERATED-Werte</b>	<b>1</b>
<b>1.1</b>	<b>DEFAULT-Werte</b>	<b>1</b>
1.1.1	CREATE TABLE und ALTER TABLE: WITH DEFAULT-Klausel	1
1.1.1.1	Default-Typen und ihre Charakteristiken	1
<b>1.2</b>	<b>GENERATED-Klausel</b>	<b>2</b>
1.2.1	CREATE TABLE und ALTER TABLE: GENERATED-Klausel	2
1.2.1.1	Überblick	2
1.2.1.2	Unterstützte Charakteristiken und Daten-Typen	2
1.2.1.3	Behandlung von GENERATED ALWAYS und GENERATED BY DEFAULT	3
<b>1.3</b>	<b>Beispiel für GENERATED ALWAYS (über CREATE TABLE)</b>	<b>4</b>
1.3.1	CREATE TABLE- und INSERT-Beispiele	4
1.3.2	Besonderheiten bei Utilities	6
1.3.2.1	UNLOAD-Utility-Beispiel	6
1.3.2.2	LOAD-Utility-Beispiel	7
1.3.2.2.1	Grundsätzliches Verhalten beim LOAD	7
1.3.2.2.2	Besondere Bedingungen beim LOAD für GENERATED ALWAYS-Spalten	7
1.3.2.2.3	LOAD-Beispiele	7
1.3.2.2.3.1	Laden der UNLOAD-Daten mit REPLACE	7
1.3.2.2.3.2	Laden der UNLOAD-Daten mit REPLACE und IDENTITYOVERRIDE	8
1.3.3	Katalog-Tabellen-Infos für DEFAULT- und GENERATED ALWAYS	9
<b>1.4</b>	<b>Beispiel für GENERATED BY DEFAULT (über CREATE TABLE)</b>	<b>10</b>
1.4.1	CREATE TABLE- und INSERT	10
1.4.2	Besonderheiten bei Utilities	12
1.4.2.1	UNLOAD-Utility-Beispiel	12
1.4.2.2	LOAD-Utility-Beispiel	13
1.4.2.2.1	Grundsätzliches Verhalten beim LOAD	13
1.4.2.2.2	Besondere Bedingungen beim LOAD für GENERATED BY DEFAULT-Spalten	13
1.4.2.2.3	LOAD-Beispiele	13
1.4.2.2.3.1	Laden der UNLOAD-Daten mit REPLACE	13
1.4.3	Katalog-Tabellen-Infos für DEFAULT- und GENERATED BY DEFAULT	14
<b>1.5</b>	<b>Beispiel für GENERATED ALWAYS (über ALTER TABLE)</b>	<b>15</b>
1.5.1	Szenario der Spalten-Definition	15
1.5.2	Katalog-Tabellen-Infos für DEFAULT- und GENERATED-Werte nach ALTER TABLE	17
<b>1.6</b>	<b>Zusammenfassung der Parametrisierungs-Effekte</b>	<b>18</b>
1.6.1	Regeln für die GENERATED-Klausel	18
1.6.2	Detail-Zusammenhänge der Parameter beim LOAD	19

# 1 DEFAULT- und GENERATED-Werte

## 1.1 DEFAULT-Werte

### 1.1.1 CREATE TABLE und ALTER TABLE: WITH DEFAULT-Klausel

#### 1.1.1.1 Default-Typen und ihre Charakteristiken

Beim Insert bzw. LOAD von Daten und in bestimmten Fällen auch beim Update kann DB2 einen Default-Wert einsetzen. Dies geschieht durch die WITH DEFAULT-Klausel.

Die Default-Werte-Behandlung kann bei der Definition einer Spalte im Rahmen von CREATE TABLE oder ALTER TABLE spezifiziert werden.

Es gelten folgende Grund-Regeln:

- Wenn weder DEFAULT noch NOT NULL vorgegeben wurden, wird die Spalte als NULL-fähig definiert. Dann ist der Default-Wert ein NULL-Wert.
- Werden außer diesem Parameter keine weiteren Default-Parameter vorgegeben, setzt DB2 automatisch in Abhängigkeit vom Datentyp Default-Werte ein. Dies erfolgt beim INSERT oder LOAD implizit.

Wird eine Tabelle nachträglich mit ALTER ADD COLUMN nachträglich erweitert und dieser Spalte wird WITH DEFAULT zugewiesen, ist beim UPDATE eine abweichende Behandlung für die existierenden Zeilen zu beachten.

Die folgende Tabelle zeigt die Default-Werte-Zuordnung für INSERT und LOAD sowie für UPDATE einer Zeile nach dem Einfügen einer neuen Spalte in eine Tabelle mit existierenden Daten:

Beim INSERT oder LOAD werden die Default-Werte eingesetzt (bei einem UPDATE findet keine Veränderung statt- Ausnahme: AS ROW CHANGE TIMESTAMP):		Beim UPDATE .. (nach ALTER ADD column)
Daten-Typ	Default-Wert-Implizit	Default-Wert-Implizit
Numerisch	0	0
String mit fester Länge	Blank oder hexa Nullen (Binär)	Blank oder hexa Nullen (Binär)
String mit variabler Länge	Ein String mit der Länge 0	Ein String mit der Länge 0
DATE	CURRENT DATE	01.01.0001
TIME	CURRENT TIME	00.00.00
TIMESTAMP	CURRENT TIMESTAMP	01.01.0001.00.00.00-000000
Distinct Typ	Der Default-Wert des Source-Daten-Typs, sofern keine explizite Default-Maßnahme.	Der Default-Wert des Source-Daten-Typs, sofern keine explizite Default-Maßnahme.

  

Spalten-Charakteristik	Default-Wert-Zuweisung	Default-Wert-Zuweisung
constant	Spezifizierter Wert, z.B. 1 oder 'F'	Spezifizierter Wert, z.B. 1 oder 'F'
NULL	NULL-Wert	NULL-Wert
CURRENT SQLID	Inhalt Spezialregister CURRENT SQLID zum Ausführungszeitpunkt.	Inhalt Spezialregister CURRENT SQLID zum Zeitpunkt des ALTER TABLE-Statements.
SESSION_USER oder USER	Inhalt Spezialregister SESSION_USER zum Ausführungszeitpunkt.	Inhalt Spezialregister SESSION_USER zum Zeitpunkt des ALTER TABLE-Statements.
ROWID GENERATED ALWAYS oder GENERATED BY DEFAULT	Vergabe eines ROWID-Werts. Änderung durch nachfolgenden REORG möglich.	Vergabe eines ROWID-Werts. Änderung durch nachfolgenden REORG möglich.
GENERATED ALWAYS oder GENERATED BY DEFAULT AS IDENTITY	Vergabe einer lfd. Nr. Wird ggf. bei LOAD verändert.	REORG-Pending-Status. Vergabe der lfd. Nr. durch REORG.
GENERATED ALWAYS oder GENERATED BY DEFAULT FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP	CURRENT TIMESTAMP. Bei jeder Änderung wird dieser Wert automatisch durch CURRENT TIMESTAMP ersetzt.	AREO* - Advimore REORG-Pending-Status. Vergabe des CURRENT TIMESTAMPS durch REORG.

Bei einem ROWID- oder einem XML-Daten-Typ sowie bei einer IDENTITY- oder Row Change Timestamp-Charakteristik darf dieser Parameter nicht vorgegeben werden.

Bei einem LOB-Spalten-Typ darf nur DEFAULT NULL vorgegeben werden.

## 1.2 GENERATED-Klausel

### 1.2.1 CREATE TABLE und ALTER TABLE: GENERATED-Klausel

#### 1.2.1.1 Überblick

Mit der GENERATED-Klausel kann ein Automatismus im DB2 zur Vergabe von Werten genutzt werden. Grundsätzlich sind die Optionen unterstützt:

- **GENERATED ALWAYS**                      DB2 generiert bei INSERT und LOAD einen Wert.
- **GENERATED BY DEFAULT**              DB2 generiert bei INSERT und LOAD nur dann einen Wert, sofern der Benutzer keine Überschreibung vornimmt.

#### 1.2.1.2 Unterstützte Charakteristiken und Daten-Typen

Die Klausel GENERATED beim CREATE TABLE oder ALTER TABLE kann angewandt werden für:

Charakteristik	Bedeutung	GENERATED ALWAYS	GENERATED BY DEFAULT
<b>IDENTITY</b>	Vergabe einer lfd. Nr. durch DB2. Nutzbar für alle numerischen Ganzzahlen-Daten-Typen.	Die Nr. wird immer durch DB2 vergeben. Besondere Beachtung ist beim UNLOAD/LOAD erforderlich, sonst werden die Nr. beim Laden neu vergeben. Beim REORG wird nur dann eine Nr. vergeben, wenn zuvor mit ALTER TABLE eine neue Spalte mit dieser Charakteristik bei bestehenden Datenzeilen aufgenommen wurde.	Die Nr. kann durch den Benutzer vergeben werden. Wird die Spalte bei einer Manipulation nicht vorgegeben, vergibt DB2 die Nr. automatisch. Hier können sich evtl. von Benutzern vergebene Nr. als problematisch erweisen, wenn sie nicht der strikten Nummernvergabe-Logik folgen.
<b>ROWID</b>	Vergabe einer ROWID, die sich aus einem eindeutigen Identifikator und der physischen Zeilen-Adresse zusammensetzt (RID). Nutzbar nur in Verbindung mit dem Daten-Typ ROWID.	Die ROWID wird immer durch DB2 vergeben. Beim Laden erfolgt eine Neu-Vergabe der Nr. Beim REORG kann sich der RID-Teil verändern.	Diese Option sollte nur genutzt werden, wenn sichergestellt ist, dass die vom Benutzer vorgegebene Nr. vorher von DB2 vergeben wurde.
<b>ROW CHANGE TIMESTAMP</b>	Vergabe eines Änderungs-Zeitpunkts. Nutzbar nur in Verbindung mit dem Daten-Typ TIMESTAMP.	Der letzte Änderungszeitpunkt wird immer durch DB2 vergeben. Beim Laden erfolgt eine Neu-Vergabe des Änderungszeitpunkts. Beim REORG wird nur dann eine Nr. vergeben, wenn zuvor mit ALTER TABLE eine neue Spalte mit dieser Charakteristik bei bestehenden Datenzeilen aufgenommen wurde.	Der Änderungszeitpunkt kann durch den Benutzer vergeben werden. Wird die Spalte bei einer Manipulation nicht vorgegeben, vergibt DB2 den Änderungszeitpunkt automatisch. Hier können sich evtl. von Benutzern vergebene Änderungszeitpunkt als problematisch erweisen, wenn sie einer nicht nachvollziehbaren Logik folgen.

### 1.2.1.3 Behandlung von GENERATED ALWAYS und GENERATED BY DEFAULT

<b>GENERATED</b>	<p>Konstante zur Spezifikation, dass DB2 in Abhängigkeit von weiteren Charakteristiken einen Wert für die Spalte generieren soll.</p> <p>Diese Option muss vorgegeben werden bei IDENTITY und ROW CHANGE TIMESTAMP. Die folgenden Parameter sind relevant in Verbindung mit dieser GENERATED-Klausel und der gewählten Option:</p> <ul style="list-style-type: none"> <li>- LOAD-UTILITY:             <ul style="list-style-type: none"> <li>- Parameter: IGNOREFIELDS und IDENTITYOVERRIDE.</li> </ul> </li> <li>- INSERT-Statement             <ul style="list-style-type: none"> <li>- Klauseln: OVERRIDING USER VALUES und DEFAULT</li> </ul> </li> <li>- MERGE-Statement             <ul style="list-style-type: none"> <li>- Klausel: DEFAULT</li> </ul> </li> <li>- UPDATE-Statement             <ul style="list-style-type: none"> <li>- Klausel: DEFAULT</li> </ul> </li> </ul>
<b>ALWAYS</b>	<p>DB2 generiert immer einen Wert für die Spalte. Diese Option wird bei den unterstützten Typen grundsätzlich empfohlen, außer bei Data Propagation (Verteilung und Austausch von Daten mit Generierung von Ids durch Fremdsysteme).</p> <p>Es ist aber zu beachten, dass auch beim LOAD grundsätzlich automatisch neue Werte generiert werden. Dies ist i.d.R. unerwünscht bei IDENTITY- und ROW CHANGE TIME-Stamp-Werten. Für IDENTITY-Spalten gibt es ab DB2-Version 9 eine Option beim Laden, die diese Problematik lösen hilft (LOAD Option IDENTITYOVERRIDE).</p> <p>Es gelten folgende Regeln:</p> <ul style="list-style-type: none"> <li>- IDENTITY Ein Unique Index auf der einzelnen Spalte ist dann erforderlich, wenn die Eindeutigkeit gewährleistet werden soll. Es ist zu beachten, dass speziell bei der Definition von CYCLE bei Erreichen der Grenzwerte bereits vergebene Nr. neu vergeben werden können. Diese sind bei vorhandenem Unique Index vorher zu löschen.</li> <li>- ROWID Bei einer ROWID ist dies der Default-Wert. DB2 generiert den ROWID-Wert in Abhängigkeit von der physischen Adresse. Ein Unique Index ist nicht erforderlich.</li> <li>- ROW CHANGE TIMESTAMP Der Zeitpunkt der Einfügung oder Änderung einer Zeile wird automatisch eingestellt. Auch beim LOAD wird der Wert neu vergeben.</li> </ul>
<b>BY DEFAULT</b>	<p>DB2 generiert einen Wert beim Einfügen und Laden einer Zeile, sofern der Wert nicht bereits gefüllt übergeben wird.</p> <p>Wird beim LOAD die Spalte definiert und ein entsprechender Wert vorgegeben, erfolgt die Übernahme des Wertes aus dem Eingabebestand.</p> <p>Beim ROWID-Spalten-Typ muss es sich um eine gültige und von DB2 vergebene ROWID handeln.</p> <p>Es gelten folgende Regeln:</p> <ul style="list-style-type: none"> <li>- IDENTITY Grundsätzlich analog ALWAYS. Aber es ist zu beachten, dass bei diesem Verfahren z.B. eine hohe Nr. vergeben werden kann. Danach schließt DB2 die Lücke, sofern die Default-Maßnahme wirkt. Und bei Erreichen des einmal vorher vergebenen Wertes wird ein doppelter Key gemeldet!</li> <li>- ROWID DB2 übernimmt einen vorher von DB2 generierten ROWID-Wert. Ein Unique Index ist für die ROWID-Spalte erforderlich. Vor der Existenz dieses Indexes ist kein Einfügen oder Laden von Zeilen möglich. Wenn zum Zeitpunkt des CREATEs Standard-Regeln gelten (CURRENT RULES = 'STD'), legt DB2 automatisch diesen Index an. Namenskonvention:              lccccccccrrrrrrr l = Index              ccccccccc = Spalten-Name, evtl mit "_" aufgefüllt              rrrrrr = generierte Zufalls-Zeichen.</li> <li>- ROW CHANGE TIMESTAMP Der Zeitpunkt der Einfügung oder Änderung einer Zeile wird nur dann automatisch eingestellt, wenn der Benutzer keine Vorgabe tätigt.</li> </ul>



## 1.3 Beispiel für GENERATED ALWAYS (über CREATE TABLE)

### 1.3.1 CREATE TABLE- und INSERT-Beispiele

Die folgenden Beispiele zeigen die grundsätzliche Verhaltensweise der Default-Werte-Behandlung beim INSERT auf:

```
CREATE TABLE SEMTYPD
( SEMCODE CHAR (15) NOT NULL
, DAUER DECIMAL (5,1) WITH DEFAULT 1.0 NOT NULL
, USER CHAR (10) WITH DEFAULT SESSION_USER NOT NULL
, SQLID CHAR (10) WITH DEFAULT CURRENT_SQLID NOT NULL
, DATE DATE WITH DEFAULT NOT NULL
, TIME TIME WITH DEFAULT NOT NULL
, TS TIMESTAMP WITH DEFAULT NOT NULL
, RCTS_ALWAYS TIMESTAMP NOT NULL
  GENERATED ALWAYS FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP
, ROWID_ALWAYS ROWID NOT NULL
  GENERATED ALWAYS
, IDENTITY_ALWAYS SMALLINT NOT NULL
  GENERATED ALWAYS AS IDENTITY
, TITEL VARCHAR (60) -- Variabler String, max. 60 Bytes
)
IN SEMDB01.SEMTS01 -- Zuordnung zur Database SEMDB01
-- Tablespace SEMTS01
;

INSERT INTO SEMTYPD ( SEMCODE ) -- RCT_ALWAYS-Generierung: 2009-02-28-23.33.00.926594
VALUES ( 'DB2-Design' ) ;
INSERT INTO SEMTYPD ( SEMCODE ) -- RCT_ALWAYS-Generierung: 2009-02-28-23.33.01.115917
VALUES ( 'DB2-Prog ' ) ;

SELECT *
FROM SEMTYPD ;
+-----+-----+-----+-----+-----+-----+-----+
! SEMCODE ! DAUER ! USER ! SQLID ! DATE ! TIME ! TS !
+-----+-----+-----+-----+-----+-----+-----+
1_! DB2-Design ! 1.0 ! DENNE ! DENNE ! 2009-02-28 ! 23.33.00 ! 2009-02-28-23.33.00.923842
2_! DB2-Prog ! 1.0 ! DENNE ! DENNE ! 2009-02-28 ! 23.33.01 ! 2009-02-28-23.33.01.115904
+-----+-----+-----+-----+-----+-----+-----+

PAGE 2
+-----+-----+-----+-----+-----+-----+-----+
! RCTS_ALWAYS ! ROWID_ALWAYS ! IDENTITY_ALWAYS !
+-----+-----+-----+-----+-----+-----+-----+
1_! 2009-02-28-23.33.00.926594 ! a:t .q ! 1 !
! ! 87A81B32600F90000000004444444444444444444444 ! !
! ! 1A3A8C22941A8010000021000000000000000000000 ! !
! ! ! ! !
2_! 2009-02-28-23.33.01.115917 ! .q ! 2 !
! ! 34181B31600F90000000004444444444444444444444 ! !
! ! 023A8C26941A8010000022000000000000000000000 ! !
! ! ! ! !
+-----+-----+-----+-----+-----+-----+-----+

PAGE 3
+-----+-----+-----+-----+-----+-----+-----+
! TITEL !
+-----+-----+-----+-----+-----+-----+-----+
1_! ? !
2_! ? !
+-----+-----+-----+-----+-----+-----+-----+
SUCCESSFUL RETRIEVAL OF 2 ROW(S)
```

\*\*\*INPUT STATEMENT:

```
SET CURRENT SQLID = 'HUBER'  
;
```

```
INSERT INTO SEMTPD ( SEMCODE ) -- RCT_ALWAYS-Generierung: 2009-02-28-23.33.01.163124  
VALUES ( 'DB2-Perf ' ) -- SQLID = 'HUBER'  
;
```

```
UPDATE SEMTPD -- RCT_ALWAYS-Änderung: 2009-02-28-23.33.01.183330  
SET TITEL = 'DB2-Design logisches und phys..' -- SQLID der Einfügung wird gehalten.  
WHERE SEMCODE = 'DB2-Design'  
;
```

Nach einigen Minuten wurde vorgegeben:

```
INSERT INTO SEMTPD ( SEMCODE , TITEL ) -- RCT_ALWAYS-Generierung: 2009-02-28-23.52.44.440086  
VALUES ( 'XML-DB2 ', 'XML und DB2 - Grundlagen ' )  
;
```

```
SELECT *  
FROM SEMTPD  
;
```

```
+-----+  
! SEMCODE ! DAUER ! USER ! SQLID ! DATE ! TIME ! TS !  
+-----+  
1_! DB2-Design ! 1.0 ! DENNE ! DENNE ! 2009-02-28 ! 23.33.00 ! 2009-02-28-23.33.00.923842  
2_! DB2-Prog ! 1.0 ! DENNE ! DENNE ! 2009-02-28 ! 23.33.01 ! 2009-02-28-23.33.01.115904  
3_! DB2-Perf ! 1.0 ! DENNE ! HUBER ! 2009-02-28 ! 23.33.01 ! 2009-02-28-23.33.01.162941  
4_! XML-DB2 ! 1.0 ! DENNE ! DENNE ! 2009-02-28 ! 23.52.44 ! 2009-02-28-23.52.44.435522  
+-----+
```

PAGE 2

```
+-----+  
! RCTS_ALWAYS ! ROWID_ALWAYS ! IDENTITY_ALWAYS !  
+-----+  
1_! 2009-02-28-23.33.01.183330 ! a:t .q ! 1 !  
! ! 87A81B32600F900000000044444444444444444444 ! !  
! ! 1A3A8C22941A8010000021000000000000000000 ! !  
! ! ! ! ! !  
2_! 2009-02-28-23.33.01.115917 ! .q ! 2 !  
! ! 34181B31600F900000000004444444444444444444 ! !  
! ! 023A8C26941A8010000022000000000000000000 ! !  
! ! ! ! ! !  
3_! 2009-02-28-23.33.01.163124 ! *^ .q ! 3 !  
! ! 55181B31600F900000000004444444444444444444 ! !  
! ! CF3A8C2C941A8010000023000000000000000000 ! !  
! ! ! ! ! !  
4_! 2009-02-28-23.52.44.440086 ! .q ! 4 !  
! ! D9011B30600F900000000004444444444444444444 ! !  
! ! BAA98C20941A8010000024000000000000000000 ! !  
! ! ! ! ! !  
+-----+
```

PAGE 3

```
+-----+  
! TITEL !  
+-----+  
1_! DB2-Design logisches und phys.. !  
2_! ? !  
3_! ? !  
4_! XML und DB2 - Grundlagen !  
+-----+
```

SUCCESSFUL RETRIEVAL OF 4 ROW(S)





## 1.3.2 Besonderheiten bei Utilities

### 1.3.2.1 UNLOAD-Utility-Beispiel

Beim UNLOAD ergeben sich keine Besonderheiten für die DEFAULT-Werte.  
Für alle GENERATED ALWAYS-Spalten werden aber Dummy-Namen generiert, die mit DSN beginnen.

Vorgabe für UNLOAD:

```
LISTDEF LISTE
  INCLUDE TABLESPACES
    TABLESPACE SEMDB01.SEMTS01
UNLOAD LIST LISTE
  PUNCHDDN (SPUNCH)
  UNLDDN (UNLDDN)
```

PUNCH-Ausgabe von UNLOAD:

Als Spalten-Namen werden vom UNLOAD-Utility bei Spalten, die mit GENERATED ALWAYS definiert sind, generiert:

<b>DSN_IDENTITY</b>	<b>für IDENTITY-Spalten.</b>
<b>DSN_ROWID</b>	<b>für ROWID-Daten-Typen.</b>
<b>DSN_RCTIMESTAMP</b>	<b>für ROW CHANGE TIMESTAMP-Spalten.</b>

Bei Auftreten zumindest einer Spalte mit der Definition GENERATED ALWAYS, generiert UNLOAD ebenfalls die Klausel:

**IGNOREFIELDS YES**

Ausgabe des Utility-Laufs:

```
TEMPLATE B8241HFD
  DSN ('DENNE.GT9G.D107.SEMTS01.U2009060.Z0628')
  DISP (OLD, KEEP, KEEP)
LOAD DATA INDDN B8241HFD LOG NO RESUME YES
  UNICODE CCSID(00367,01208,01200)
  SORTKEYS 4
  INTO TABLE "DENNE"."SEMTPD"
  WHEN (00001:00002) = X'0039'
  IGNOREFIELDS YES
( "SEMCODE" POSITION( 00003:00017) CHAR MIXED(015)
, "DAUER" POSITION( 00018:00020) DECIMAL
, "USER" POSITION( 00021:00030) CHAR MIXED(010)
, "SQLID" POSITION( 00031:00040) CHAR MIXED(010)
, "DATE" POSITION( 00041:00050) DATE EXTERNAL
, "TIME" POSITION( 00051:00058) TIME EXTERNAL
, "TS" POSITION( 00059:00084) TIMESTAMP EXTERNAL
, "DSN_RCTIMESTAMP" POSITION( 00085:00110) TIMESTAMP EXTERNAL
, "DSN_ROWID" POSITION( 00111:00152) ROWID
, "DSN_IDENTITY" POSITION( 00153:00154) SMALLINT
, "TITEL" POSITION( 00156:00217) VARCHAR MIXED
  NULLIF(00155)=X'FF'
)
```

### 1.3.2.2 LOAD-Utility-Beispiel

#### 1.3.2.2.1 Grundsätzliches Verhalten beim LOAD

Die generierten UNLOAD-Kontroll-Informationen müssen vor einem LOAD modifiziert werden:

- RESUME YES            Lädt die Daten zusätzlich zu den vorhandenen Daten.  
Da die Test-Tabelle über einen Primary Index auf der Spalte SEMCODE verfügt, werden sämtliche Zeilen mit dem Fehler "Verstoß gegen UNIQUE-Constraint" zurückgewiesen.

#### 1.3.2.2.2 Besondere Bedingungen beim LOAD für GENERATED ALWAYS-Spalten

Bei Einsatz des LOAD-Utilities gelten besondere Bedingungen für Spalten, die mit der GENERATED ALWAYS-Klausel spezifiziert wurden.

Zwei Parameter steuern die Behandlungs- und Vorgabe-Möglichkeit der Original-Spalten-Namen einer Tabelle:

- **IGNOREFIELDS**    Spezifikation, ob LOAD Spalten-Namen übergehen soll, die zwischen LOAD-Control- und Table-Definition differieren.
  - NO                    Die Spalten-Namen dürfen nicht differieren. Dies ist der Default-Wert.
  - YES                  Die Spalten-Namen dürfen differieren. DB2 ignoriert die entsprechenden Werte.
- **IDENTITYOVERRIDE** Spezifikation, ob entladene IDENTITY-Spalten, die mit GENERATED ALWAYS definiert sind, mit ihren entladenen Werten geladen werden sollen und keine automatische Vergabe der Nr. durch DB2 erfolgen soll.  
Mit dieser Option kann auch ein LOAD INTO TABLE PART nnnn vorgenommen werden, wenn die IDENTITY-Spalte Teil des Partitioning Keys ist.

#### 1.3.2.2.3 LOAD-Beispiele

##### 1.3.2.2.3.1 Laden der UNLOAD-Daten mit REPLACE

Das folgende Beispiel zeigt das erste Zurückladen der Entlade-Daten mit den generierten Optionen, aber mit

- Ersatz von RESUME YES in REPLACE

Datenzustand nach LOAD:

```

SEMCODE          DAUER USER          SQLID          DATE          TIME
-----
DB2-Design       1.0 DENNE          DENNE          2009-02-28 23.33.00
DB2-Prog         1.0 DENNE          DENNE          2009-02-28 23.33.01
DB2-Perf         1.0 DENNE          HUBER          2009-02-28 23.33.01
XML-DB2          1.0 DENNE          DENNE          2009-02-28 23.52.44

TS                RCTS_ALWAYS
-----
2009-02-28-23.33.00.923842 2009-03-01-01.00.00.709688
2009-02-28-23.33.01.115904 2009-03-01-01.00.00.710253
2009-02-28-23.33.01.162941 2009-03-01-01.00.00.710378
2009-02-28-23.52.44.435522 2009-03-01-01.00.00.710501

ROWID_ALWAYS          IDENTITY_ALWAYS  TITEL
-----
DDF7271198BC3233690401FA980001000100000000201          5 DB2-Design logische
D64F271198BC3201690401FA980001000100000000202          6 ?
174F271198BC3211690401FA980001000100000000203          7 ?
26CF271198BC3220690401FA980001000100000000204          8 XML und DB2 - Grund
***** Bottom of Data *****
    
```

Es ist zu erkennen, dass alle DEFAULT-Werte, wie DAUER, USER, SQLID, DATE, TIME und TS gehalten werden.

Für alle GENERATED ALWAYS-Spalten wurden neue Werte generiert.



### 1.3.2.2.3.2 Laden der UNLOAD-Daten mit REPLACE und IDENTITYOVERRIDE

Das folgende Beispiel zeigt das erste Zurückladen der Entlade-Daten mit den generierten Optionen, aber mit

- Ersatz von RESUME YES in REPLACE
- Vorgabe der IDENTITYOVERRIDE-Option zur Übernahme der entladenen IDENTITY-Werte.

Erster Versuch zeigt die Fehlermeldung:

```
SNU323I  -GT9G 060 01:13:10.64 DSNURWUF - COLUMN 'IDENTITY_ALWAYS' IS OMITTED
```

Heißt: Der Dummy-Name DSN\_IDENTITY muss auf den konkreten Namen verändert werden.

Zweiter Versuch mit folgenden LOAD-Optionen:

```
LOAD DATA INDDN B8241HFD LOG NO NOCOPYPEND REPLACE
UNICODE CCSID(00367,01208,01200)
  SORTDEVT SYSDA WORKDDN (WORKDDN1 , WORKDDN2)
SORTKEYS                4  IDENTITYOVERRIDE
INTO TABLE "DENNE"."SEMTPD"
WHEN(00001:00002) = X'0039'
IGNOREFIELDS YES
( "SEMCODE"  POSITION( 00003:00017) CHAR MIXED(015)
, "DAUER"    POSITION( 00018:00020) DECIMAL
, "USER"     POSITION( 00021:00030) CHAR MIXED(010)
, "SQLID"    POSITION( 00031:00040) CHAR MIXED(010)
, "DATE"     POSITION( 00041:00050) DATE EXTERNAL
, "TIME"     POSITION( 00051:00058) TIME EXTERNAL
, "TS"       POSITION( 00059:00084) TIMESTAMP EXTERNAL
, "DSN_RCTIMESTAMP" POSITION( 00085:00110) TIMESTAMP EXTERNAL
, "DSN_ROWID"   POSITION( 00111:00152) ROWID
, "IDENTITY_ALWAYS" POSITION( 00153:00154) SMALLINT
, "TITEL"     POSITION( 00156:00217) VARCHAR MIXED
                NULLIF(00155)=X'FF'
)
```

Datenzustand nach LOAD:

SEMCODE	DAUER	USER	SQLID	DATE	TIME
DB2-Design	1.0	DENNE	DENNE	2009-02-28	23.33.00
DB2-Prog	1.0	DENNE	DENNE	2009-02-28	23.33.01
DB2-Perf	1.0	DENNE	HUBER	2009-02-28	23.33.01
XML-DB2	1.0	DENNE	DENNE	2009-02-28	23.52.44

TS	RCTS_ALWAYS
2009-02-28-23.33.00.923842	2009-03-01-01.18.52.455697
2009-02-28-23.33.01.115904	2009-03-01-01.18.52.456261
2009-02-28-23.33.01.162941	2009-03-01-01.18.52.456384
2009-02-28-23.52.44.435522	2009-03-01-01.18.52.456507

ROWID_ALWAYS	IDENTITY_ALWAYS	TITEL
F0C05A3398BC3212690401FA98000100020000000201	1	DB2-Design logische
22A05A3398BC3209690401FA98000100020000000202	2	?
03A05A3398BC3232690401FA98000100020000000203	3	?
5C605A3398BC3239690401FA98000100020000000204	4	XML und DB2 - Grund

\*\*\*\*\* Bottom of Data \*\*\*\*\*

Es ist zu erkennen, dass alle DEFAULT-Werte, wie DAUER, USER, SQLID, DATE, TIME und TS gehalten werden. Für die GENERATED ALWAYS-Spalten, außer IDENTITY wurden neue Werte generiert.

Will man die entladenen ROW CHANGE-Werte auch erhalten und versucht, dies mit dem konkreten Spalten-Namen zu initiieren, erhält folgende Fehlermeldung:

```
DSNU269I  -GT9G 060 01:27:12.41 DSNURWUF - FIELD RCTS_ALWAYS IS NOT ALLOWED
```

**Fazit: Wir brauchen auch eine ROWCHANGEOVERRIDE-Option!!!**

### 1.3.3 Katalog-Tabellen-Infos für DEFAULT- und GENERATED ALWAYS

Der folgende Auszug stammt aus den Katalog-Tabellen SYSCOLUMNS und zeigt Details zu den DEFAULT- und GENERATED-Optionen auf.

Die DEFAULT-Spalte zeigt die DEFAULT-Definition der Spalte und DEFAULT\_WERT den Inhalt, der bei INSERT oder LOAD eingefügt wird.

Die folgenden Informationen basieren auf den Definitionen mit **GENERATED ALWAYS**:

```

+-----+
!      COLNAME      ! COLTYPE  ! LENGTH  ! NACHK  ! NULL  ! DEFAULT  ! DEFAULT_WERT
+-----+
1_! SEMCODE          ! CHAR     ! 15      !         ! N      ! N        !
2_! DAUER            ! DECIMAL  ! 5       !         ! N      ! 3        ! 1.0
3_! USER             ! CHAR     ! 10      !         ! N      ! U        !
4_! SQLID            ! CHAR     ! 10      !         ! N      ! S        !
5_! DATE             ! DATE     ! 4       !         ! N      ! Y        !
6_! TIME             ! TIME     ! 3       !         ! N      ! Y        !
7_! TS               ! TIMESTMP ! 10      !         ! N      ! Y        !
8_! RCTS_ALWAYS     ! TIMESTMP ! 10      !         ! N      ! E        !
9_! ROWID_ALWAYS    ! ROWID    ! 17      !         ! N      ! A        !
10_! IDENTITY_ALWAYS ! SMALLINT ! 2       !         ! N      ! I        !
11_! TITEL           ! VARCHAR  ! 60      !         ! Y      ! Y        !
+-----+
    
```



## 1.4 Beispiel für GENERATED BY DEFAULT (über CREATE TABLE)

### 1.4.1 CREATE TABLE- und INSERT

Die folgenden Beispiele zeigen die grundsätzliche Verhaltensweise der Default-Werte-Behandlung beim INSERT auf:

```
CREATE TABLE SEMTPD
( SEMCODE CHAR (15) NOT NULL
, DAUER DECIMAL (5,1) WITH DEFAULT 1.0 NOT NULL
, USER CHAR (10) WITH DEFAULT SESSION_USER NOT NULL
, SQLID CHAR (10) WITH DEFAULT CURRENT_SQLID NOT NULL
, DATE DATE WITH DEFAULT NOT NULL
, TIME TIME WITH DEFAULT NOT NULL
, TS TIMESTAMP WITH DEFAULT NOT NULL
, RCTS_DEFAULT TIMESTAMP NOT NULL
GENERATED BY DEFAULT FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP
, ROWID_DEFAULT ROWID NOT NULL
GENERATED BY DEFAULT
, IDENTITY_DEFAULT SMALLINT NOT NULL
GENERATED BY DEFAULT AS IDENTITY
, TITEL VARCHAR (60) -- Variabler String, max. 60 Bytes
)
IN SEMDB01.SEMTS01 -- Zuordnung zur Database SEMDB01
-- Tablespace SEMTS01
;
INSERT INTO SEMTPD ( SEMCODE )
VALUES ( 'DB2-Design' ) ;
```

```
DSNT408I SQLCODE = -540, ERROR: THE DEFINITION OF TABLE DENNE.SEMTPD IS INC
A REQUIRED UNIQUE INDEX
DSNT418I SQLSTATE = 57001 SQLSTATE RETURN CODE
```

Eine Spalte mit Daten-Typ ROWID, die mit GENERATED BY DEFAULT definiert wurde, fordert einen Unique Index:

```
-- Anlegen Unique Index fuer ROWID-Spalte
CREATE UNIQUE INDEX SEMTPD_UIRID
ON SEMTPD (ROWID_DEFAULT)
;

INSERT INTO SEMTPD ( SEMCODE ) -- RCT_DEFAULT-Generierung: 2009-03-01-03.40.35.003763
VALUES ( 'DB2-Design' ) ;
INSERT INTO SEMTPD ( SEMCODE ) -- RCT_DEFAULT-Generierung: 2009-03-01-03.40.35.334633
VALUES ( 'DB2-Prog ' ) ;

SELECT *
FROM SEMTPD ;
-----+-----
! SEMCODE ! DAUER ! USER ! SQLID ! DATE ! TIME ! TS !
-----+-----
1! DB2-Design ! 1.0 ! DENNE ! DENNE ! 2009-03-01 ! 03.40.34 ! 2009-03-01-03.40.34.999194
2! DB2-Prog ! 1.0 ! DENNE ! DENNE ! 2009-03-01 ! 03.40.35 ! 2009-03-01-03.40.35.334505
-----+-----

PAGE 2
-----+-----
! RCTS_DEFAULT ! ROWID_DEFAULT ! IDENTITY_DEFAULT!
-----+-----
1! 2009-03-01-03.40.35.003763 ! +r TQ .q ! 1 !
! ! 492EDB31600F9000000004444444444444444444444 ! !
! ! E9438C2C941A8010000021000000000000000000000 ! !
! ! ! ! !
2! 2009-03-01-03.40.35.334633 ! vmTQ .q ! 2 !
! ! 1A9EDB32600F9000000000444444444444444444444444 ! !
! ! 75438C23941A8010000022000000000000000000000 ! !
! ! ! ! !
-----+-----
SUCCESSFUL RETRIEVAL OF 2 ROW(S)
```



```

***INPUT STATEMENT:
  SET CURRENT SQLID = 'HUBER'
  ;

INSERT INTO SEMTYPD ( SEMCODE )           -- RCT_DEFAULT-Generierung: 2009-03-01-03.40.35.460750
VALUES ( 'DB2-Perf ' )                   -- SQLID = 'HUBER'
;

UPDATE SEMTYPD                            -- RCT_DEFAULT-Änderung: 2009-03-01-03.40.35.478515
SET TITEL = 'DB2-Design logisches und phys..' -- SQLID der Einfügung wird gehalten.
WHERE SEMCODE = 'DB2-Design'
;

```

Nach einigen Minuten wurde vorgegeben:

```

INSERT INTO SEMTYPD ( SEMCODE , TITEL )   - RCT_DEFAULT-Generierung: 2009-03-01-03.52.53.107779
VALUES ( 'XML-DB2 ' , 'XML und DB2 - Grundlagen ' )
;

UPDATE SEMTYPD                            -- RCT_DEFAULT-Änderung: 3999-12-31-23.59.59.000000
SET RCTS_DEFAULT = TIMESTAMP ( '39991231235959' )
where semcode = 'DB2-Prog '
;

SELECT *
FROM SEMTYPD
;

```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! SEMCODE ! DAUER ! USER ! SQLID ! DATE ! TIME ! TS !
+-----+-----+-----+-----+-----+-----+-----+-----+
1! DB2-Design ! 1.0 ! DENNE ! DENNE ! 2009-03-01 ! 03.40.34 ! 2009-03-01-03.40.34.999194
2! DB2-Prog ! 1.0 ! DENNE ! DENNE ! 2009-03-01 ! 03.40.35 ! 2009-03-01-03.40.35.334505
3! DB2-Perf ! 1.0 ! DENNE ! HUBER ! 2009-03-01 ! 03.40.35 ! 2009-03-01-03.40.35.460623
4! XML-DB2 ! 1.0 ! DENNE ! DENNE ! 2009-03-01 ! 03.52.53 ! 2009-03-01-03.52.53.103459
+-----+-----+-----+-----+-----+-----+-----+-----+

```

PAGE 2

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! RCTS_DEFAULT ! ROWID_DEFAULT ! IDENTITY_DEFAULT!
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1! 2009-03-01-03.40.35.478515 ! +r TQ .q ! 1 !
! ! ! 492EDB31600F9000000000444444444444444444444444444444 ! !
! ! ! E9438C2C941A8010000021000000000000000000000 ! !
! ! ! ! ! ! ! ! ! !
2! 3999-12-31-23.59.59.000000 ! vmTQ .q ! 2 !
! ! ! 1A9EDB32600F9000000000444444444444444444444444444444 ! !
! ! ! 75438C23941A8010000022000000000000000000000000000 ! !
! ! ! ! ! ! ! ! ! !
3! 2009-03-01-03.40.35.460750 ! MTQ .q ! 3 !
! ! ! B4DEDB30600F9000000000444444444444444444444444444444 ! !
! ! ! 19438C2C941A80100000230000000000000000000000000000 ! !
! ! ! ! ! ! ! ! ! !
4! 2009-03-01-03.52.53.107779 ! b D Q .q ! 4 !
! ! ! 83CCDB30600F9000000000444444444444444444444444444444 ! !
! ! ! 244F8C23941A80100000240000000000000000000000000000 ! !
! ! ! ! ! ! ! ! ! !
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

PAGE 3

```

+-----+-----+-----+-----+
! TITEL !
+-----+-----+-----+-----+
1! DB2-Design logisches und phys.. !
2! ? !
3! ? !
4! XML und DB2 - Grundlagen !
+-----+-----+-----+-----+

```

SUCCESSFUL RETRIEVAL OF 4 ROW(S)

## 1.4.2 Besonderheiten bei Utilities

### 1.4.2.1 UNLOAD-Utility-Beispiel

Beim UNLOAD ergeben sich keine Besonderheiten für die DEFAULT-Werte.

Für alle GENERATED BY DEFAULT-Spalten werden die Original-Namen und keine Dummy-Namen generiert.

#### Vorgabe für UNLOAD:

```
LISTDEF LISTE
  INCLUDE TABLESPACES
    TABLESPACE SEMDB01.SEMTS01
UNLOAD LIST LISTE
  PUNCHDDN (SPUNCH)
  UNLDDN (UNLDDN)
```

#### PUNCH-Ausgabe von UNLOAD:

Da alle Spalten mit DEFAULT-bzw. GENERATED BY DEFAULT definiert sind, werden die Original-Namen erzeugt. Die Klausel IGNOREFIELDS wird nicht generiert.

#### Ausgabe des Utility-Laufs:

```
TEMPLATE B83D8C5K
  DSN ('DENNE.GT9G.D107.SEMTS01.U2009060.Z0958')
  DISP (OLD,KEEP,KEEP)
LOAD DATA INDDN B83D8C5K LOG NO RESUME YES
UNICODE CCSID(00367,01208,01200)
SORTKEYS 8
INTO TABLE "DENNE"."SEMTPD"
WHEN(00001:00002) = X'0039'
( "SEMCODE"
  POSITION( 00003:00017) CHAR MIXED(015)
, "DAUER"
  POSITION( 00018:00020) DECIMAL
, "USER"
  POSITION( 00021:00030) CHAR MIXED(010)
, "SQLID"
  POSITION( 00031:00040) CHAR MIXED(010)
, "DATE"
  POSITION( 00041:00050) DATE EXTERNAL
, "TIME"
  POSITION( 00051:00058) TIME EXTERNAL
, "TS"
  POSITION( 00059:00084) TIMESTAMP EXTERNAL
, "RCTS_DEFAULT"
  POSITION( 00085:00110) TIMESTAMP EXTERNAL
, "ROWID_DEFAULT"
  POSITION( 00111:00152) ROWID
, "IDENTITY_DEFAULT"
  POSITION( 00153:00154) SMALLINT
, "TITEL"
  POSITION( 00156:00217) VARCHAR MIXED
  NULLIF(00155)=X'FF'
)
```

**1.4.2.2 LOAD-Utility-Beispiel**  
**1.4.2.2.1 Grundsätzliches Verhalten beim LOAD**

Die generierten UNLOAD-Kontroll-Informationen müssen vor einem LOAD modifiziert werden:

- RESUME YES            Lädt die Daten zusätzlich zu den vorhandenen Daten.  
                           Da die Test-Tabelle über einen Primary Index auf der Spalte SEMCODE verfügt,  
                           werden sämtliche Zeilen mit dem Fehler "Verstoß gegen UNIQUE-Constraint"  
                           zurückgewiesen.

**1.4.2.2.2 Besondere Bedingungen beim LOAD für GENERATED BY DEFAULT-Spalten**

Bei Einsatz des LOAD-Utilities gelten keine besonderen Bedingungen für Spalten, die mit der GENERATED-Klausel spezifiziert wurden.

**1.4.2.2.3 LOAD-Beispiele**  
**1.4.2.2.3.1 Laden der UNLOAD-Daten mit REPLACE**

Das folgende Beispiel zeigt das erste Zurückladen der Entlade-Daten mit den generierten Optionen, aber mit

- Ersatz von RESUME YES in REPLACE

Datenzustand nach LOAD:

SEMCODE	DAUER	USER	SQLID	DATE	TIME
DB2-Design	1.0	DENNE	DENNE	2009-03-01	03.40.34
DB2-Prog	1.0	DENNE	DENNE	2009-03-01	03.40.35
DB2-Perf	1.0	DENNE	HUBER	2009-03-01	03.40.35
XML-DB2	1.0	DENNE	DENNE	2009-03-01	03.52.53

  

TS	RCTS_DEFAULT
2009-03-01-03.40.34.999194	2009-03-01-03.40.35.478515
2009-03-01-03.40.35.334505	3999-12-31-23.59.59.000000
2009-03-01-03.40.35.460623	2009-03-01-03.40.35.460750
2009-03-01-03.52.53.103459	2009-03-01-03.52.53.107779

  

ROWID_DEFAULT	IDENTITY_DEFAULT	TITEL
4E9924E3D8BC321C690401FA98000100010000000201	1	DB2-Design logische
17A594E3D8BC3223690401FA98000100010000000202	2	?
B149D4E3D8BC320C690401FA98000100010000000203	3	?
8234C4CFD8BC3203690401FA98000100010000000204	4	XML und DB2 - Grund

\*\*\*\*\* Bottom of Data \*\*\*\*\*

Es ist zu erkennen, dass alle DEFAULT-Werte, wie DAUER, USER, SQLID, DATE, TIME und TS gehalten werden.

Auch alle GENERATED BY DEFAULT-Spalten wurden mit ihren entladenen Werten übernommen.



### 1.4.3 Katalog-Tabellen-Infos für DEFAULT- und GENERATED BY DEFAULT

Der folgende Auszug stammt aus den Katalog-Tabellen SYSCOLUMNS und zeigt Details zu den DEFAULT- und GENERATED-Optionen auf.

Die DEFAULT-Spalte zeigt die DEFAULT-Definition der Spalte und DEFAULT\_WERT den Inhalt, der bei INSERT oder LOAD eingefügt wird.

Die folgenden Informationen basieren auf den Definitionen mit **GENERATED BY DEFAULT**:

```

+-----+
!      COLNAME      ! COLTYPE !  LENGTH  !  NACHK  !  NULL  ! DEFAULT ! DEFAULT_WERT
+-----+
1_! SEMCODE          ! CHAR   !    15   !         ! 0 ! N   ! N         !
2_! DAUER             ! DECIMAL !     5   !         ! 1 ! N   ! 3         ! 1.0
3_! USER              ! CHAR   !    10   !         ! 0 ! N   ! U         !
4_! SQLID             ! CHAR   !    10   !         ! 0 ! N   ! S         !
5_! DATE              ! DATE   !     4   !         ! 0 ! N   ! Y         !
6_! TIME              ! TIME   !     3   !         ! 0 ! N   ! Y         !
7_! TS                ! TIMESTMP !    10   !         ! 0 ! N   ! Y         !
8_! RCTS_DEFAULT      ! TIMESTMP !    10   !         ! 0 ! N   ! F         !
9_! ROWID_DEFAULT     ! ROWID   !    17   !         ! 0 ! N   ! D         !
10_! IDENTITY_DEFAULT ! SMALLINT !     2   !         ! 0 ! N   ! J         !
11_! TITEL             ! VARCHAR !    60   !         ! 0 ! Y   ! Y         !
+-----+

```

## 1.5 Beispiel für GENERATED ALWAYS (über ALTER TABLE)

### 1.5.1 Szenario der Spalten-Definition

Zunächst wurde mit der Definition der Tabelle mit einer einzigen Spalte begonnen:

```
CREATE TABLE SEMTYPD
  ( SEMCODE CHAR (15) NOT NULL
  )
  IN SEMDB01.SEMTS01 ;
```

Danach wurde 'Zug-um-Zug' eine Spalte nach der anderen eingefügt.

Dazwischen erfolgten INSERT-Statements, die langsam die Spalten-Entwicklungen dokumentierten. Beispiel (COMMITTS sind wg. Wechsel zwischen DDL und DML erforderlich):

```
COMMIT ;
INSERT INTO SEMTYPD ( SEMCODE )
  VALUES ( 'DB2-ADES ' ) ;
COMMIT ;
ALTER TABLE SEMTYPD
  ADD COLUMN DAUER DECIMAL (5,1) WITH DEFAULT ;
COMMIT ;
INSERT INTO SEMTYPD ( SEMCODE )
  VALUES ( 'DB2-BDES ' ) ;
```

```
SELECT * FROM SEMTYPD ;
+-----+
! SEMCODE ! DAUER !
+-----+
1_! DB2-ADES ! 1.0 !
2_! DB2-BDES ! 1.0 !
+-----+
```

SUCCESSFUL RETRIEVAL OF 2 ROW(S)  
COMMIT ;

```
ALTER TABLE SEMTYPD
  ADD COLUMN
  USER CHAR (10) WITH DEFAULT SESSION_USER NOT NULL ;
COMMIT ;
INSERT INTO SEMTYPD ( SEMCODE )
  VALUES ( 'DB2-CDES ' ) ;
```

```
SELECT * FROM SEMTYPD ;
+-----+
! SEMCODE ! DAUER ! USER !
+-----+
1_! DB2-ADES ! 1.0 ! DENNE !
2_! DB2-BDES ! 1.0 ! DENNE !
3_! DB2-CDES ! 1.0 ! DENNE !
+-----+
```

SUCCESSFUL RETRIEVAL OF 3 ROW(S)

Nach Definition der ROW CHANGE TIMESTAMP-Charakteristik mit ALTER TABLE ADD COLUMN wird der Status AREO\* auf Tablespace-Ebene gesetzt, der einen REORG empfiehlt, bei dem die Änderungs-Zeitstempel auf Zeilenebene vergeben werden:

```
-DIS DB(SEMDB01) SPACENAM(SEMTS01)
```

NAME	TYPE PART	STATUS	PHYERRLO	PHYERRHI	CATALOG	PIECE
SEMTS01	TS	RW,AREO*				



Nach Definition der IDENTITY-Spalte mit ALTER TABLE ADD COLUMN wird der Status REORP auf Tablespace-Ebene gesetzt, der einen REORG fordert, bei dem die ROWIDs auf Zeilenebene vergeben werden:

```
DSNT404I SQLCODE = 610, WARNING: A CREATE/ALTER ON OBJECT SEMDB01.SEMTYPD_PI HAS PLACED OBJECT IN REBUILD PENDING
DSNT418I SQLSTATE = 01566 SQLSTATE RETURN CODE
```

```
-DIS DB(SEMDB01) SPACENAM(SEMTS01)
NAME        TYPE PART  STATUS      PHYERRLO PHYERRHI CATALOG  PIECE
-----
SEMST01    TS          RW,REORP
```

Jeder Zugriff wird nun abgewiesen mit:

```
DSNT408I SQLCODE = -904, ERROR: UNSUCCESSFUL EXECUTION CAUSED BY AN UNAVAILABLE RESOURCE. REASON 00C900AD, TYPE OF RESOURCE 00000200, AND RESOURCE NAME SEMDB01.SEMTS01
```

Nach zwischenzeitlichem REORG (der die IDENTITY-Werte einfügt) ergab sich folgendes Gesamt-Bild der Zeilen (vor jeder Zeile wurde ja eine weitere Spalte hinzugefügt, was die zunehmenden Default-Zuweisungen erklärt):

```
SELECT * FROM SEMTYPD ;
```

	SEMCODE	DAUER	USER	SQLID	DATE	TIME	TS
1_!	DB2-ADES	1.0	DENNE	DENNE	0001-01-01	00.00.00	0001-01-01-00.00.00.000000
2_!	DB2-BDES	1.0	DENNE	DENNE	0001-01-01	00.00.00	0001-01-01-00.00.00.000000
3_!	DB2-CDES	1.0	DENNE	DENNE	0001-01-01	00.00.00	0001-01-01-00.00.00.000000
4_!	DB2-DDES	1.0	DENNE	DENNE	0001-01-01	00.00.00	0001-01-01-00.00.00.000000
5_!	DB2-EDES	1.0	DENNE	DENNE	2009-03-01	00.00.00	0001-01-01-00.00.00.000000
6_!	DB2-FDES	1.0	DENNE	DENNE	2009-03-01	09.04.29	0001-01-01-00.00.00.000000
7_!	DB2-GDES	1.0	DENNE	DENNE	2009-03-01	09.04.29	2009-03-01-09.04.29.732260
8_!	DB2-HDES	1.0	DENNE	DENNE	2009-03-01	09.04.29	2009-03-01-09.04.29.811899
9_!	DB2-IDES	1.0	DENNE	DENNE	2009-03-01	09.04.29	2009-03-01-09.04.29.893659

  

	RCTS_ALWAYS	ROWID_ALWAYS	IDENTITY_ALWAYS
1_!	2009-03-01-08.47.00.828884	4000800003000000000444444444444444444444	1
!	!	0000001609210000021000000000000000000000	!
2_!	2009-03-01-08.47.00.828884	4000400003000000000444444444444444444444	2
!	!	0000001609210000022000000000000000000000	!
3_!	2009-03-01-08.47.00.828884	ä	3
!	!	4000C00003000000000444444444444444444444	!
!	!	0000001609210000023000000000000000000000	!
4_!	2009-03-01-08.47.00.828884		4
!	!	4000200003000000000444444444444444444444	!
!	!	0000001609210000024000000000000000000000	!
5_!	2009-03-01-08.47.00.828884		5
!	!	4000A00003000000000444444444444444444444	!
!	!	0000001609210000025000000000000000000000	!
6_!	2009-03-01-08.47.00.828884	-	6
!	!	4000600003000000000444444444444444444444	!
!	!	0000001609210000026000000000000000000000	!
7_!	2009-03-01-08.47.00.828884	ö	7
!	!	4000E00003000000000444444444444444444444	!
!	!	0000001609210000027000000000000000000000	!
8_!	2009-03-01-09.04.29.814775		8
!	!	4000I000030000000000444444444444444444444	!
!	!	0000001609210000028000000000000000000000	!
9_!	2009-03-01-09.04.29.896286	/ .q	9
!	!	26BB0B30600F9000000000444444444444444444	!
!	!	41124C27941A8010000290000000000000000000	!

Nun wurde noch einmal ein UPDATE durchgeführt, der die Auswirkung auf die ROW CHANGE TIME-STAMP-Spalte aufzeigen soll:

```
SET DAUER = 2
where semcode = 'DB2-ADES '
```

```

+-----+
| SEMCODE | DAUER | USER | SQLID | DATE | TIME | TS |
+-----+
1 | DB2-ADES | 2.0 | DENNE | DENNE | 0001-01-01 | 00.00.00 | 0001-01-01-00.00.00.000000
+-----+

+-----+
| RCTS_ALWAYS | ROWID_ALWAYS | IDENTITY_ALWAYS |
+-----+
1 | 2009-03-01-09.08.21.128315 | 4000800003000000000044444444444444444444 | 1
| | 0000001609210000002100000000000000000000 |
| |
| |
+-----+

```

1.5.2 Katalog-Tabellen-Infos für DEFAULT- und GENERATED-Werte nach ALTER TABLE

```

+-----+
| COLNAME | COLTYPE | LENGTH | NACHK | NULL | DEFAULT | DEFAULT_WERT |
+-----+
1 | SEMCODE | CHAR | 15 | 0 | N | N |
2 | DAUER | DECIMAL | 5 | 1 | N | 3 | 1.0
3 | USER | CHAR | 10 | 0 | N | U | DENNE
4 | SQLID | CHAR | 10 | 0 | N | S | DENNE
5 | DATE | DATE | 4 | 0 | N | Y |
6 | TIME | TIME | 3 | 0 | N | Y |
7 | TS | TIMESTMP | 10 | 0 | N | Y |
8 | RCTS_ALWAYS | TIMESTMP | 10 | 0 | N | E |
9 | ROWID_ALWAYS | ROWID | 17 | 0 | N | A |
10 | IDENTITY_ALWAYS | SMALLINT | 2 | 0 | N | I |
+-----+

```



## 1.6 Zusammenfassung der Parametrisierungs-Effekte

### 1.6.1 Regeln für die GENERATED-Klausel

Folgende Regeln können abgeleitet werden:

- **GENERATED ALWAYS**
  - Eine Technik, bei der immer die Schlüssel neu vergeben werden, ist praktisch sehr fragwürdig. PKs werden neu vergeben und FKs passen nicht mehr! Vorsicht auch aber bei der Überleitung von Daten aus einem System in ein anderes mit UNLOAD-LOAD-Techniken Die Nummernkreise sind i.d.R. nicht synchronisiert und führen zu Konflikten. Dies kann nur in speziellen Fällen akzeptiert werden! Ab der DB2-Version 9 kann für IDENTITY-Spalten die IDENTITYOVERRIDE-Klausel beim LOAD verwendet werden.
  - Wer im Ausnahmefall immer automatisch generierte Daten benötigt oder akzeptieren kann, ist mit DEFAULT- und GENERATED ALWAYS-Definitionen gut bedient. Die Unterstützung von immer wieder neu generierten Schlüsseln ist durch UNLOAD gewährleistet. Dort werden anstelle der konkreten Spalten-Namen Dummy-Namen mit DSN\_ beginnend sowie die Klausel IGNOREFIELDS YES generiert. Der nachfolgende LOAD ignoriert dann diese Eingabedaten. Alternativ können in diesem Fall die Spalten-Deklarationen für solche GENERATED ALWAYS-Spalten ganz weggelassen werden.
  - Wer nur die IDENTITY-Charakteristik nutzen will, kann bei GENERATED ALWAYS ab der DB2-Version 9 mit Hilfe der IDENTITYOVERRIDE-Klausel die bestehenden Daten bewahren. Aber: es muss in die LOAD-Control-Statements eingegriffen werden. Leider kann man beim UNLOAD diese Option nicht für die automatische PUNCH-Ausgabe ansteuern.
  - Wer die neue ROW CHANGE TIMESTAMP-Charakteristik nutzen will, kann bei GENERATED ALWAYS kein Pendant zur IDENTITYOVERRIDE-Klausel finden. Das heißt, in diesem Fall generiert DB2 beim LOAD immer den Ladezeitpunkt als letzten Änderungszeitpunkt. Damit kann dieses Konzept nur als technische Unterstützung zur Erkennung der physisch letzten Veränderung herangezogen werden, z.B. zur Kontrolle auf zwischenzeitlichen Parallel-Update.
  - Für einen ROWID-Daten-Typ ist GENERATED ALWAYS die empfohlene Option, da dieser Wert i.d.R. ausschließlich internen Aufgabenstellungen (gedacht für die Beziehungen von LOB-Objekten zu den Basis-Objekten) dient.
- **GENERATED BY DEFAULT**
  - Eine Technik, bei der die Schlüssel nur dann neu vergeben werden, wenn keine Vorgabe durch den Benutzer erfolgt, kann praktisch ebenfalls fragwürdig sein, da damit unerwartete Ergebnisse auftreten können. Wenn z.B. ein IDENTITY-Wert manuell außerhalb des Vergabe-Nummernkreises oder weit außerhalb des aktuellen Standes vergeben wird, drohen nachfolgende Probleme.
  - Wer immer die einmal automatisch generierten Daten behalten will oder muss, kann dies in einer Kombination von mit DEFAULT- und GENERATED BY DEFAULT-Definitionen sowie Zugriffsbeschränkungen z.B. über Trigger oder Views lösen. Die Unterstützung der Übernahme vorhandener Schlüssel ist beim UNLOAD und LOAD ohne spezielle Eingriffe gewährleistet. UNLOAD generiert bei GENERATED BY DEFAULT-Spalten die Original-Spalten-Namen. LOAD akzeptiert dann die Benutzervorgabe und die automatische Generierung wird ausgesetzt.
  - Als Alternative zu einer IDENTITY-Charakteristik mit GENERATED BY DEFAULT kann ab DB2-Version 8 eine SEQUENCE genutzt werden. Der nächste oder vorhergehende Wert muss mit den Funktionen NEXT VALUE FOR bzw. PREVIOUS VALUE FOR angefordert werden. Das LOAD-Utility kann ohne Besonderheiten eingesetzt werden.

### 1.6.2 Detail-Zusammenhänge der Parameter beim LOAD

Die Zusammenhänge der Parameter werden nach entsprechend intensiven (und nervigen) Tests in der nachfolgenden Tabelle dargestellt:

Charakteristik	GENERATED	IGNORE-FIELDS	IDENTITYOVER-RIDE	Spalten-Namen	Bemerkung
IDENTITY	ALWAYS	NO	Nicht vorgegeben	Original-Spalten-Namen	Abweisung mit Fehlermeldung "IDENTITY Spalte ist nicht erlaubt".
				DSN_IDENTITY	Abweisung mit Fehlermeldung "DSN_IDENTITY Spalte wurde nicht gefunden".
			Vorgegeben	Original-Spalten-Namen	Daten werden geladen. Übernahme der entladenen Nr.
				DSN_IDENTITY	Abweisung mit Fehlermeldung "IDENTITY Spalte wurde nicht vorgegeben".
		YES	Nicht vorgegeben	DSN_IDENTITY	Daten werden geladen. Nr. wird neu generiert.
				Original-Spalten-Namen	Abweisung mit Fehlermeldung "Spalten-Namen nicht erlaubt".
	Vorgegeben		Original-Spalten-Namen	Daten werden geladen. Übernahme der entladenen Nr.	
			DSN_IDENTITY	Abweisung mit Fehlermeldung "IDENTITY Spalte wurde nicht vorgegeben".	
	BY DEFAULT	NO	Nicht vorgegeben	Original-Spalten-Namen	Daten werden geladen. Übernahme der entladenen Nr.
				Vorgegeben	Original-Spalten-Namen
		YES	Nicht vorgegeben	Original-Spalten-Namen	Daten werden geladen. Übernahme der entladenen Nr.
				Vorgegeben	Original-Spalten-Namen
		-	-	DSN_IDENTITY	Abweisung mit Fehlermeldung "Spalte nicht gefunden".
	ROWID	ALWAYS	NO	Egal (nicht relevant)	Original-Spalten-Namen
YES				Egal (nicht relevant)	DSN_ROWID
			Egal (nicht relevant)	Original-Spalten-Namen	Abweisung mit Fehlermeldung "Spalten-Namen nicht erlaubt".
BY DEFAULT		wird nicht empfohlen und wurde nicht weiter getestet.			
ROW CHANGE TIME-STAMP		ALWAYS	NO	Egal (nicht relevant)	Original-Spalten-Namen
	YES			Nicht vorgegeben	DSN_RCTIMESTAMP
			Egal (nicht relevant)	Original-Spalten-Namen	Abweisung mit Fehlermeldung "Spalten-Namen nicht erlaubt".
	BY DEFAULT	NO	Egal (nicht relevant)	Original-Spalten-Namen	Daten werden geladen. Übernahme der entladenen Timestamp-Werte.
				DSN_RCTIMESTAMP	Abweisung mit Fehlermeldung "Spalte nicht gefunden".
		YES	Egal (nicht relevant)	DSN_RCTIMESTAMP	Daten werden geladen. Generierung neuer Timestamp-Werte.
			Original-Spalten-Namen	Daten werden geladen. Übernahme der entladenen Timestamp-Werte.	