

Willkommen zum „IBM DB2 Newsletter“

Liebe Leserinnen und Leser,

Die Mondfinsternis, mit Blutmond und die Sonnenfinsternis, 2 Naturereignisse in einem Jahr, was für ein Spektakel. Für kurze Zeit (erdgeschichtlich) verschwanden diese Objekte.

Ich weiß das Sie sich freuen, wenn der DB2 Newsletter erscheint und Sie haben sicherlich bemerkt, das die Häufigkeit des Erscheinens über die Zeit abgenommen hat. Damit der Newsletter aber nicht auf Dauer verschwindet, brauche ich IHRE Hilfe in Form von fertigen Beiträgen (technischer Art) oder auch nur Ideen, Stickpunkte,



Für Fragen und Anregungen unsere Kontaktadresse: db2news@de.ibm.com.

Ihr TechTeam

Inhaltsverzeichnis

DB2 NL 01/2015 – NACHTRAG – FEHLERBEREINIGUNG.....	2
SEQUENCES.....	2
ARCHIVIERUNG DB2 DIAGNOSE LOGFILE.....	6
BACKUP INCLUDE LOG – BETRACHTUNGEN.....	6
CHATS MIT DEM LABOR.....	6
SCHULUNGEN / TAGUNGEN / INFORMATIONSVERANSTALTUNG.....	7
DB2 REGIONALE USERGROUP	7
DB2 SERVICES ONLINE ZUM VORZUGSPREIS BESTELLEN.....	7
NEWSLETTER ARCHIV.....	7
ANMELDUNG/ABMELDUNG.....	7
DIE AUTOREN DIESER AUSGABE.....	8

DB2 NL 01/2015 – Nachtrag – Fehlerbereinigung

Im Artikel „Stolpersteine ...“ gab es einen Stolperstein in der Tabelle mit den HADR-Status. Hier die korrekte Version:

	Seite L	Seite R
HADR-Status	PRIMARY	STANDBY
Sonntag	Full-Backup	
Montag/Dienstag	Incremental	
Mittwoch	CRASH	Takeover
HADR-Status	-	PRIMARY
Mittwoch Abend		Incremental
Aufbau der DB	Incremental Restore in neue DB: restore <TS_INCR> restore <TS_FULL> - - - ERROR , falsches Full-Backup - - -	

Sequences

Vor einiger Zeit erhielt ich folgende Anfrage:

Nach einem LOAD in eine Tabelle mit einer Column GENERATED ALWAYS AS IDENTITY und MODIFIED BY IDENTITYIGNORE

benötigt man den nächsten Wert für diese Spalte. Mit dem db2look erhält man diesen Wert in der Form:

```
ALTER TABLE ... ALTER COLUMN ... RESTART WITH <VALUE>.
```

Frage: Wie kann man per SQL diesen Wert abfragen?

Da Sequenzen immer wieder vorkommen und Mr. Google ein Unmenge an Informationen zusammenbringt, wollte ich hier nun mal die Erkenntnisse zusammenbringen und natürlich gleich auch testen.

Als Beispiel nehme ich eine kleine Spieltabelle, mit folgendem Inhalt:

```
db2v101@db2blu:~> db2 "select * from TEST.TABELLE"
SPALTE1 SPALTE2 SPALTE4
```

```
-----
1 A 1
2 B 2
3 C 3
4 D 4
```

Um den Restart-Wert für eine Sequence herauszubekommen gibt es verschiedene Möglichkeiten:

- Abfrage mittels db2look für die entsprechende Tabelle.

```
db2look -d MYDB -e -t TABELLE -s TEST
```

```
db2v101@db2blu:~> db2look -d MYDB -e -t TABELLE -z TEST
-- No userid was specified, db2look tries to use Environment variable USER
-- USER is: DB2V101
-- Specified SCHEMA is: TEST
```

```
...
-- This CLP file was created using DB2LOOK Version "10.1"
..
CONNECT TO MYDB;
-----
-- DDL Statements for Table "TEST"."TABELLE"
-----
```

```
CREATE TABLE "TEST"."TABELLE" (
  "SPALTE1" INTEGER ,
  "SPALTE2" VARCHAR(30) ,
  "SPALTE4" INTEGER NOT NULL GENERATED ALWAYS AS IDENTITY (
```

```

START WITH +1
INCREMENT BY +1
MINVALUE +1
MAXVALUE +2147483647
NO CYCLE
NO CACHE
NO ORDER ) )
IN "WLMMONITOR" ;

```

```
ALTER TABLE "TEST"."TABELLE" ALTER COLUMN "SPALTE4" RESTART WITH 5;
```

```

COMMIT WORK;
CONNECT RESET;
TERMINATE;

```

• Abfrage auf den System Katalog

Dazu habe ich folgende Abfragen gefunden (zugesendet bekommen)

○ Abfrage des letzten gesetzten Wertes

```

select c.tbcreator, c.tbname, c.name, s.increment, s.start, s.maxvalue, s.minvalue, s.cycle,
s.cache, s.order , s.LASTASSIGNEDVAL from SYSIBM.SYSCOLUMNS as c inner join
SYSIBM.SYSDEPENDENCIES as d on d.dname = c.tbname and d.dschem = c.tbcreator inner join
SYSIBM.SYSSEQUENCES as s on s.seqname = d.bname and s.seqschem = d.bschem where c.identity
= 'Y' and d.dtype = 'T' order by c.tbcreator, c.tbname, c.name

```

```

TBCREATOR TBNAME NAME INCREMENT START MAXVALUE MINVALUE CYCLE CACHE ORDER LASTASSIGNEDVAL
-----
TABELLE SPALTE4 1. 1. 2147483647. 1. N 1 N 4. -----TEST

```

○ Abfrage des nächsten Wertes (gecached) wenn NO CACHE, dann nächster Wert.

```

select c.tabschema, c.tabname, c.colname, d.increment, d.start, d.maxvalue, d.minvalue,
d.cycle, d.cache, d.NEXTCACHEFIRSTVALUE from SYSCAT.COLUMNS as c inner join
SYSCAT.COLIDENTATTRIBUTES as d on d.tabname = c.tabname and d.tabschem = c.tabschem and
d.colname = c.colname where c.identity = 'Y' order by c.tabschem, c.tabname, c.colname

```

```

TABSHEMA TABNAME COLNAME INCREMENT START MAXVALUE MINVALUE CYCLE CACHE NEXTCACHEFIRSTVALUE
-----
TEST TABELLE SPALTE4 1. 1. 2147483647. 1. N 1 5.

```

○ Eine weitere Variante um die aktuelle Sequence-Nummer herauszubekommen ist db2cat. Mit nachfolgendem SQL kann man den db2cat Befehl zusammenstellen lassen:

```

WITH temp1 ( alias ) AS ( SELECT CURRENT SERVER FROM sysibm.sysdummy1) SELECT 'db2cat -p
sequence -d ' || TRIM(alias) || ' -s ' || TRIM(SEQSHEMA) || ' -n ' || TRIM(SEQNAME) || ' ; '
FROM SYSIBM.SYSDEPENDENCIES a, SYSIBM.SYSSEQUENCES b, syscat.columns c, temp1 WHERE
b.SEQSHEMA NOT LIKE 'SYS%' AND b.SEQTYPE = 'I' AND a.BNAME = b.SEQNAME AND a.DNAME =
c.TABNAME AND a.DSCHEMA = c.TABSHEMA AND c.IDENTITY = 'Y'
1

```

```
db2cat -p sequence -d MYDB -s TEST -n SQL150924144038200 ;
```

Beispiel-Ausgabe:

```
db2v101@db2blu:~> db2cat -p sequence -d MYDB -s TEST -n SQL150924144038200
```

```

_____ DB2 Relational Data Services Tools _____
I B M
DB2CAT
This tool is a utility designed to help in examining packed
descriptor information stored in the system catalog tables.

```

```

DB2 Version 10.1, 5622-044 (c) Copyright IBM Corp. 2011
Licensed Material - Program Property of IBM
IBM DATABASE 2 Catalog Analysis and Repair Tool

```

Please use single quote for delimited name

```

Start Database Manager Successful.
Connecting to the Database.
Connect to Database Successful.
Binding package - Bind was Successful.
Enter outfile name. Default is to terminal ==>
DB2 Version 10.1, 5622-044 (c) Copyright IBM Corp. 2011
Licensed Material - Program Property of IBM

```

Please use single quote for delimited name

SYSSEQUENCES.SEQ_DESC: TEST.SQL150924144038200

SYSSEQUENCES.SEQ_DESC HEADER

```
Version : 3840
Length : 184
Length of creator name : 8
Creator name : DB2V101
Length of schema name : 8
Schema name : TEST
Length of Owner of SEQUENCE : 8
Owner of SEQUENCE : DB2V101
Unique sequence ID : 6
Support of temp sequences : 1
Node number : 0
The following Flags are specified:
Sequence is an identity column
Identity is generated always
Data Type {hex, type} : 0x0018 , UNKNOWN TYPE
Source Type {hex, type} : 0x0001 , INTEGER
Length of Source Type : 10
CACHE column type : 1
Creator type : U
Owner type : U
Create time : {2015-09-22-17.14.27.721072}
Alter time : {2015-09-22-17.14.27.721072}
Start with Data : +00000000000000000000000000000001.
Minvalue Data : +00000000000000000000000000000001.
Maxvalue Data : +0000000000000000000000002147483647.
Increment Data : +00000000000000000000000000000001.
Restart With Data : +000000000000000000000000000001.
```

Executing Connect Reset -- Connect Reset was Successful.

- Mit den oben gelieferten Werten lässt sich jedoch noch nicht die Sequence der Identity Spalte ändern, daher hier die Statements mit dem man die ALTER

```
SELECT 'ALTER TABLE ' || TRIM(a.DSCHEMA) || '.' || TRIM(a.DNAME) || ' ALTER COLUMN ' ||
RTRIM(c.COLNAME) || ' RESTART WITH ' || CAST(CAST(COALESCE(LASTASSIGNEDVAL+1,1) AS BIGINT) AS
CHAR(10)) || ' ; ' FROM SYSIBM.SYSDEPENDENCIES a, SYSIBM.SYSSEQUENCES b, syscat.columns c
WHERE b.SEQSCHEMA NOT LIKE 'SYS%' AND b.SEQTYPE = 'I' AND a.BNAME = b.SEQNAME AND a.DNAME =
c.TABNAME AND a.DSCHEMA = c.TABSCHEMA AND c.IDENTITY = 'Y'
1
```

ALTER TABLE TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH 5 ;

Eine andere Variante ist:

```
SELECT 'ALTER TABLE ' || RTRIM(TABSCHEMA) || '.' || RTRIM(TABNAME) || ' ALTER COLUMN ' ||
RTRIM(COLNAME) || ' RESTART WITH ' || CHAR(BIGINT(NEXTCACHEFIRSTVALUE)) || ';' FROM
SYSCAT.COLIDENTATTRIBUTES ORDER BY TABSCHEMA, TABNAME, COLNAME
1
```

ALTER TABLE TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH 5 ;

Soweit so gut, aber wie sieht das nun nach dem Load aus? Funktionieren diese Abfragen auch noch?

Ich habe dazu verschiedene Test gemacht und muss die Frage leider mit NEIN beantworten, hier mein Beispiel:

Mit folgendem LOAD Kommando habe ich zusätzliche Datensätze hinzugefügt:

```
> db2 -v "load from load.txt of del modified by coldel, identityoverride replace into TEST.TABELLE
nonrecoverable"
> cat load.txt
1,'Z',1
2,'Z',2
```

```
3,'Z',3
4,'Z',4
5,'Z',7
```

Der Inhalt der Tabelle hat sich wie folgt verändert

```
> db2 "select * from TEST.TABELLE"
SPALTE1 SPALTE2 SPALTE4
```

```
-----
1 'Z' 1
2 'Z' 2
3 'Z' 3
4 'Z' 4
5 'Z' 7
```

5 record(s) selected.

Jedoch hat sich bei der Abfrage auf die Sequence nichts geändert. Diese weist immer noch auf RESTART mit 5 hin.

```
SELECT 'ALTER TABLE ' || RTRIM(TABSCHEMA) || '.' || RTRIM(TABNAME) || ' ALTER COLUMN ' ||
RTRIM(COLNAME) || ' RESTART WITH ' || CHAR(BIGINT(NEXTCACHEFIRSTVALUE)) || ';' FROM
SYSCAT.COLIDENTATTRIBUTES ORDER BY TABSCHEMA, TABNAME, COLNAME
1
```

```
-----
ALTER TABLE TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH 5 ;
```

Auch ein runstats/reorg änderte daran nichts. Die einzige Möglichkeit, die ich gefunden habe, um dieses Problem zu lösen, ist die MAX Abfrage auf die Sequence Spalte.

Mit der Nachfolgenden Abfrage, kann erstmal der MAX Wert für die Sequence Spalte ermittelt werden:

```
Select 'select max('||c.name||') from '||rtrim(c.tbcreator)||'.'||c.tbname from SYSIBM.SYSCOLUMNS as
c inner join SYSIBM.SYSDEPENDENCIES as d on d.dname = c.tbname and d.dschem = c.tbcreator inner
join SYSIBM.SYSSEQUENCES as s on s.seqname = d.bname and s.seqschem = d.bschem where c.identity =
'Y' and d.dtype = 'T' order by c.tbcreator, c.tbname, c.name
1
```

```
-----
select max(SPALTE4) from TEST.TABELLE
```

Diesen Wert nimmt man dann für den Update der Sequence, das könnte wie folgt aussehen:

```
select 'select 'alter table ',rtrim(c.tbcreator)||'.'||c.tbname,' ALTER COLUMN ',c.name, ' RESTART
WITH ' , ' , 'max('||c.name||') from '||rtrim(c.tbcreator)||'.'||c.tbname from SYSIBM.SYSCOLUMNS as
c inner join SYSIBM.SYSDEPENDENCIES as d on d.dname = c.tbname and d.dschem = c.tbcreator inner
join SYSIBM.SYSSEQUENCES as s on s.seqname = d.bname and s.seqschem = d.bschem where c.identity =
'Y' and d.dtype = 'T' order by c.tbcreator, c.tbname, c.name
```

```
select 'alter table TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH ' , max(SPALTE4) from
TEST.TABELLE
```

Integriert man dies nun in z.B. ein kleines Shell Script, sieht das wie folgt aus:

```
> ./sequence.sh
select 'alter table TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH ' , max(SPALTE4)+1 from
TEST.TABELLE
alter table TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH 5
Ausfuehren restart_sequence.tmp.sql (Y/n) Y
alter table TEST.TABELLE ALTER COLUMN SPALTE4 RESTART WITH 5
DB20000I The SQL command completed successfully.
```

Das Skript sieht dann z.B. wie folgt aus:

```
cat ./sequence.sh
```

```
SQLFILE=restart_sequence.tmp.sql
```

```
db2 -x "select 'select 'alter table ',rtrim(c.tbcreator)||'.'||c.tbname,' ALTER COLUMN ',c.name, '
RESTART WITH ' , ' , 'max('||c.name||')+1 from '||rtrim(c.tbcreator)||'.'||c.tbname from
SYSIBM.SYSCOLUMNS as c inner join SYSIBM.SYSDEPENDENCIES as d on d.dname = c.tbname and d.dschem =
c.tbcreator inner join SYSIBM.SYSSEQUENCES as s on s.seqname = d.bname and s.seqschem = d.bschem
where c.identity = 'Y' and d.dtype = 'T' order by c.tbcreator, c.tbname, c.name" | tr -s " "
```

```

>$$SQLFILE.1

cat $$SQLFILE
#read answer?"Ausfuehren (Y/n) "
#echo "Ausfuehren (Y/n) "; read answer
answer='Y'
[ "$answer" = "Y" ] && db2 -xf $$SQLFILE.1 | tr -s " " >$$SQLFILE

head -10 $$SQLFILE
#read answer?"Ausfuehren (Y/n) "
echo "Ausfuehren $$SQLFILE (Y/n) "; read answer
[ "$answer" = "Y" ] && db2 -vf $$SQLFILE | tr -s " "

[ -f "$$SQLFILE.1" ] && rm $$SQLFILE.1

```

Natürlich müsste das Skript noch etwas angepasst werden, z.B. auf eine einzelne Sequence/Tabelle, das bleibt Ihnen aber vorbehalten.

Archivierung DB2 Diagnose Logfile

Für kleinere, wenig genutzte bzw. Datenbanken, die nicht aktiv im Maintenance Fokus stehen, ist es empfehlenswert den Datenbank Manager Parameter DIAGSIZE zu setzen. Bei der Wahl der Größe ist jedoch zu berücksichtigen, das hier nicht die Größe des einzelnen Diagnose-Files steht, sondern die Gesamtgröße aller archivierten Files. DB2 verwendet zur Archivierung maximal 20 Files, d.h. die Größe eines Logfiles ist dann 1/20 der DIAGSIZE. Vorteil dieser Variante ist, das man sich nicht um die Bereinigung kümmern muss und die Files eine maximale Größe hat. Weiterhin wirkt dieser Parameter auch für das Notify Log. Nachteil ist jedoch, das man dadurch nie genau den Zeitraum kennt, der in den Logfiles enthalten ist. Gibt es also Probleme und es wird wesentlich mehr rausgeschrieben, steigt die Wahrscheinlichkeit, das der Anfang des Problems nicht mehr aufzufinden ist, da es aus den 20 Files rausgewandert ist.

Die Diagnose Logfiles haben eine fortlaufende Nummer. Daher schaut man bei der manuellen Analyse (nicht mit dem db2diag Tool) immer nach der höchsten Nummer.

Wird DIAGSIZE verwendet, ist aber von der Verwendung des db2diag -A abzuraten, da ALLE archivierten Diag-Logfiles archiviert werden, was durchaus kurios aussieht.

```

# db2diag -A
db2diag: Moving "/db2/icd/dump/icdi/db2diag.0.log"
to "/db2/icd/dump/icdi/db2diag.0.log_2015-08-12-09.36.41"

db2diag: Moving "/db2/icd/dump/icdi/db2diag.1.log"
to "/db2/icd/dump/icdi/db2diag.1.log_2015-08-12-09.36.41"

db2diag: Moving "/db2/icd/dump/icdi/db2diag.2.log"
to "/db2/icd/dump/icdi/db2diag.2.log_2015-08-12-09.36.41"

db2diag: Moving "/db2/icd/dump/icdi/db2diag.3.log"
to "/db2/icd/dump/icdi/db2diag.3.log_2015-08-12-09.36.41"

db2diag: Moving "/db2/icd/dump/icdi/db2diag.4.log"
to "/db2/icd/dump/icdi/db2diag.4.log_2015-08-12-09.36.41"

db2diag: Moving "/db2/icd/dump/icdi/db2diag.5.log"
to "/db2/icd/dump/icdi/db2diag.5.log_2015-08-12-09.36.41"

```

Backup include log – Betrachtungen

Bei einer Datenbank mit nur einem Data Partition Member (NODE0000) ist im ONLINE Backup INCLUDE LOGS als Default gesetzt, was in der Regel sehr schön ist, da ein Restore bis zum Ende des Backups möglich ist, ohne das die Logfiles vorgehalten sein müssen.

Mit INCLUDE LOGS werden am Ende des Backups die Logfiles - die während des Backups erstellt wurden - in das Backup eingefügt.

Befinden sich die Logfiles bereits im Archiv, werden diese aus dem Archiv wieder herausgeholt (retrieve) und angefügt. Dies kann die Laufzeit eines ONLINE Backups extrem verlängern.

Wenn der Retrieve der Logfiles nicht funktioniert, wird obendrein das komplette Backup mit SQL02428 abgelehnt und vom Backup Medium gelöscht.

Chats mit dem Labor

Eine Liste der bereits durchgeführten Chats ist [hier](#) zu finden.

Die Präsentationen der Chats können dort angeschaut und heruntergeladen werden.

Schulungen / Tagungen / Informationsveranstaltung

DB2 Regionale Usergroup

Das 9. DeDUG Meeting (DB2 LUW User Group) findet am 23. Oktober 2015 in Böblingen (IBM) statt.

Agenda und Anmeldung zu diesem kostenlosen Event [hier](#).

Nähere Details zur DeDUG auf der [IDUG Webseite](#).

DB2 Services online zum Vorzugspreis bestellen

Kennen Sie schon unseren Webauftritt zum IBM Software Service Shop?

www.ibm.com/de/software-services-shop

Dort finden Sie zahlreiche Software Service Angebote, darunter auch die beiden folgenden DB2 Services:



[Datenbank Konfigurationsprüfung durch Guardium](#)



[DB2 Datenbank Health Check](#)

Besuchen Sie uns und erfahren Sie mehr Details zu unseren Angeboten.



Newsletter Archiv

Wir haben ein weiteres Archiv für den DB2 Newsletter bei

Alte Ausgaben vom DB2-NL sind nun zum Nachlesen in den Archiven zu finden von:

- [Bytec](#)
- [Cursor Software AG](#)
- [Drap](#)
- [ids-System GmbH](#)
- [Lis.Tec](#)
- [ORDIX](#)

Anmeldung/Abmeldung

Sie erhalten diesen Newsletter bis zur 3ten Ausgabe ohne Anmeldung. Wenn Sie weiterhin diesen Newsletter empfangen wollen, schicken Sie Ihre Anmeldung mit dem Subject „ANMELDUNG“ an db2news@de.ibm.com.

Die Autoren dieser Ausgabe

Sollten Sie Anfragen zu den Artikeln haben, können Sie sich entweder direkt an den jeweiligen Autor wenden oder stellen Ihre Frage über den DB2 NL, denn vielleicht interessiert ja die Antwort auch die anderen DB2 NL Leser.

Doreen Stein	Master Certified IT-Spezialist DB2 LUW, IBM Certified DB2 LUW 10.5, IBM SWG Services; Chief-Editor DB2NL, Dipl-Inf (TU). djs@de.ibm.com
--------------	--

Reviewer und Ideenlieferanten:

Dirk Fechner	IBM SWG
Peter Schurr	IBM SWG
Frank Berghofer	IBM SWG