

Willkommen zum „IBM DB2 Newsletter“

Liebe Leserinnen und Leser,

Ja, ja, lange ist es her. Der Jahreswechsel ist Geschichte, genauso wie bereits das erste Quartal in 2015. Selbst das 2te ist schon fast rum. Kennen Sie das auch? Sie denken es ist noch ganz viel Zeit und plötzlich bleibt davon nichts mehr übrig und dann im entscheidenden Moment versagt auch noch das Backup-Konzept. Sprich kurz vor der Freigabe der Ausgabe ging alles verloren. Da predige ich in Kundensituationen ein umfangreichen Backup-Restore Konzept und im eigenen Fall hilft mir die letzte Sicherung auch nicht weiter und Logfiles gibt es nicht, die vorwärts gerollt werden könnten. Da musste das Gedächtnis ran.



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

Ihr TechTeam

Inhaltsverzeichnis

STOLPERSTEINE BEIM EINSATZ VON DB2 HADR MIT ACR UND TSA.....	2
UNTERSTÜTZTE HADR-SYNCHRONISATIONSMODI BEIM EINSATZ VON TSA (TIVOLI SYSTEM AUTOMATION)	2
BACKUP UND RECOVERY	2
NUTZUNG DES AUTOMATIC-CLIENT-REROUTE-FEATURES (ACR)	3
ARBEITEN MIT ZEITFORMATEN.....	4
CHATS MIT DEM LABOR.....	5
SCHULUNGEN / TAGUNGEN / INFORMATIONSVERANSTALTUNG.....	5
DB2 REGIONALE USERGROUP	5
IBM DB2 FÜR LINUX, UNIX UND WINDOWS IM SAP UMFELD	6
DB2 SERVICES ONLINE ZUM VORZUGSPREIS BESTELLEN.....	6
NEWSLETTER ARCHIV.....	7
ANMELDUNG/ABMELDUNG.....	7
DIE AUTOREN DIESER AUSGABE.....	7

Stolpersteine beim Einsatz von DB2 HADR mit ACR und TSA

Unterstützte HADR-Synchronisationsmodi beim Einsatz von TSA (Tivoli System Automation)

Für die Synchronisation der Datenbanken stehen bei HADR prinzipiell vier verschiedene Modi zur Verfügung: SYNC, NEARSYNC, ASYNC und SUPERASYNC.

Während der SYNC-Modus am besten vor Transaktionsverlust bei gleichzeitigem Ausfall beider Seiten schützt, hat der Modus SUPERASYNC nahezu keinen Performanceeinfluss auf die Primary.

Beim Einsatz von TSA als Clustermanager sind aber nicht alle Synchronisationsarten möglich. In einem Fehlerfall führt TSA nur dann einen HADR-Takeover durch, wenn die Datenbanken sich im PEER- oder DISCONNECTED-PEER-Status befinden. Da der Peer-Status im SUPERASYNC-Modus nie erreicht wird, fällt dieser Modus schon generell weg. Aber auch der ASYNC-Modus ist beim Einsatz von TSA nicht möglich. Da bei ASYNC die primäre Seite nicht auf eine Bestätigung des Empfangs der Logeinträge durch die Standby-Seite wartet, könnten z.B. durch einen Netzwerkausfall einige Protokollsätze der Primary auf dem Weg zur Standby verloren gehen. TSA kann dadurch nicht sicherstellen, dass der Takeover ohne Datenverlust durchgeführt werden kann.

Versucht man nach Initialisierung von HADR nun mittels db2haicu TSA zu implementieren, schlägt dies mit einer entsprechenden Meldung fehl, falls ASYNC bzw. SUPERASYNC konfiguriert ist.

Aber Vorsicht: Wird der Synchronisationsmodus nach dem Aufsetzen von TSA auf ASYNC oder SUPERASYNC geändert, wird keine Fehlermeldung ausgegeben. Im Fehlerfall kann TSA dann gar keinen Takeover durchführen (insbesondere bei SUPERASYNC) oder es ist mit Datenverlust zu rechnen.

Für die Nutzung von HADR zusammen mit TSA sind also nur die Synchronisationsmethoden SYNC oder NEARSYNC möglich.

Backup und Recovery

Für größere Datenbanken kann es im Rahmen einer Backup-Recovery-Strategie aus Platz- oder Zeitgründen notwendig sein, neben Full-Backups auch Incremental- oder Delta-Backups zu erstellen. Beim Einsatz von HADR ist es sinnvoll, ein von beiden Seiten erreichbares Sicherungsmedium zu nutzen. Beim Ausfall einer Seite hat die andere somit weiterhin Zugriff auf alle Backups und archivierten Logdateien.

Je nach gerade aktiver Seite kann es also sein, dass ein Backup mal auf der einen (Seite L) und mal von der anderen (Seite R) erstellt wird. Ein Full-Backup einer Seite ist natürlich auf der anderen wiederherstellbar – schließlich wurde der HADR-Verbund auch so initial aufgebaut.

Wie steht es aber um die Wiederherstellung mit Incremental-Backups verschiedener Seiten? – Um es gleich vorweg zu nehmen, dies ist leider nicht möglich.

Beide Seiten sind recoverytechnisch eigenständige Datenbanken. Das sehen wir auch daran, dass jede Datenbank ihr eigenes Recovery-History-File (RHF) besitzt und Backup-Operationen der einen Seite nicht im RHF der anderen Seite auftauchen. Der Inhalt des RHF ist aber letztlich für die Wiederherstellung selbst nicht entscheidend. Maßgebend sind die Tablespace-Metadaten.

Bei einem Incremental-Backup werden alle Änderungen seit dem letzten Full-Backup gesichert. Bei Delta-Backups die Änderungen seit dem letzten Backup (egal welcher Typ). In den Tablespace-Metadaten ist vermerkt, welches (Full-)Backup benötigt wird, um das jeweilige Incremental- oder Delta-Backup wiederherzustellen.

Betrachten wir dabei folgendes Szenario: In einer HADR-Umgebung wird einmal pro Woche (am Sonntag) ein Full-Backup der Datenbank erstellt. An den restlichen Tagen wird jeweils ein Incremental-Backup erzeugt. Seite L sei regulär Primary und Seite R regulär Standby. Aufgrund eines Hardwarefehlers oder einer geplanten Wartung wird am Mittwoch Nachmittag ein HADR-Takeover durchgeführt, sodass Seite R nun Primary ist. Nach Plan wäre in der folgenden Nacht ein Incremental-Backup vorgesehen.

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

Dieses Incremental-Backup der Seite R lässt sich aber nicht auf Basis des letzten sonntäglichen Full-Backups von Seite L restaurieren und ist unter diesem Gesichtspunkt nutzlos.

Ein weiterer Punkt: Das Incremental-Backup (auf Seite R) sichert alle Änderungen seit dem letzten Full-Backup auf *dieser Seite*. Dieses Full-Backup liegt wahrscheinlich schon sehr lange zurück und damit wird das Incremental eher zum „light“ Full-Backup. Weiterhin ist das notwendige Full-Backup-Image vielleicht auch schon gar nicht mehr vorhanden, sodass ein Recovery mit dem neuen Incremental Backup gar nicht mehr möglich ist.

Unter diesen Aspekten ist es also empfehlenswert:

Nach einem Takeover auf der neuen Primary-Seite als nächstes auf jeden Fall ein Full-Backup durchzuführen, auch wenn die turnusmäßige Strategie ein Incremental- oder Delta-Backup vorsieht.

Nutzung des Automatic-Client-Reroute-Features (ACR)

Automatic Client Reroute ist keine Funktionalität von HADR, kann aber zusammen damit verwendet werden. Ziel ist es, dass der Client im Fehlerfall automatisch eine Verbindung zu der Alternativseite aufbauen kann und die Unterbrechungszeit möglichst kurz ist. ACR kann insbesondere dann zum Einsatz kommen, wenn keine virtuellen IP-Adressen verwendet werden können oder sollen.

Die automatische Clientweiterleitung wird aktiviert, wenn ein Kommunikationsfehler erkannt wurde oder wenn in einer HADR-Umgebung `SQL1776N` („The command cannot be issued on an HADR standby database.“) zurückgegeben wird.

Deshalb ist die Verwendung von ACR auch nicht zusammen mit der Read-On-Standby-Funktionalität möglich, da dann ein Connect zur Standby-Seite möglich ist und kein `SQL1776N` ausgelöst wird.

Serverseitig werden für beide HADR-Partner mit dem „update alternate server“-Kommando die Alternativseiten kreuzweise konfiguriert. Während des Verbindungsaufbaus zur Datenbank erhält der Client automatisch die Parameter (Host und Port) für die Alternativseite und kann diese Information für den Neuaufbau im Falle eines Fehlers nutzen.

Auch JDBC-Treiber (Typ 2 und Typ 4) unterstützen diese Funktionalität. Bei der Nutzung eines Typ-4-Treibers sollten zusätzlich auch die JDBC-Properties

clientRerouteAlternateServerName und clientRerouteAlternatePortNumber explizit gesetzt werden. Die dort angegebenen Werte dienen vor allem dem initialen Connect und werden dann mit serverseitigen Werten beim Verbindungsaufbau überschrieben. Das Setzen der JDBC-Properties stellt jedoch sicher, dass nach einem Takeover der Datenbank (die reguläre Standby-Seite wird zum Primary) auch nach einem Neustart der Applikation bzw. des Applikationsservers der Verbindungsaufbau zur Datenbank möglich ist. Sind die Properties nicht gesetzt, kann in einem solchen Fall keine Verbindung zur regulären Seite aufgenommen werden und damit können auch nicht die Werte für Alternativserver und -port übermittelt werden.

Arbeiten mit Zeitformaten

Ich hatte die Aufgabe von einer Tabelle deren Spaltenname auf einen Zeitstempel hinwies, jedoch der Datentyp BIGINT war, die Zeitstempel-Daten zu ermitteln.

Nach der Internet Recherche zu dieser Spalte und dem entsprechenden Tool, zeigte sich, dass in dem BIGINT die Millisekunden seit dem 1.1.1970 enthalten sind.

Nun bestand die Frage, wie man aus diesem BIGINT einen Zeitstempel erhält. Mister Google hatte da auch ein paar Antworten, jedoch ein Großteil der Abfragen, funktionierten nicht für DB2. Daher habe ich hier ein paar Möglichkeiten zusammengetragen, um diese Daten auswerten kann.

Aber zuerst mal ein kleines Konstrukt, um die SQL auszuprobieren:

```
db2blu1@db2blu:~> db2 -tvf convert.table.sql
create table DB2NL.convert (when_changed bigint)
DB20000I The SQL command completed successfully.
```

```
db2blu1@db2blu:~> db2 import from convert_data.txt of del replace into DB2NL.convert
SQL3109N The utility is beginning to load data from file "convert_data.txt".
SQL3110N The utility has completed processing. "79" rows were read from the
input file.
SQL3221W ...Begin COMMIT WORK. Input Record Count = "79".
SQL3222W ...COMMIT of any database changes was successful.
SQL3149N "79" rows were processed from the input file. "79" rows were
successfully inserted into the table. "0" rows were rejected.
Number of rows read = 79
Number of rows skipped = 0
Number of rows inserted = 79
Number of rows updated = 0
Number of rows rejected = 0
Number of rows committed = 79
```

```
db2blu1@db2blu:~> head -3 convert_data.txt
1428919534238
1429524708040
1429524708044
```

Zu berücksichtigen ist, in welcher Zeitzone die Daten gespeichert wurden. So muss z.B. in der Sommerzeit in Deutschland 2 Stunden auf GMT addiert werden.

Im nachfolgenden Beispiel habe ich in Spalte 1 den BIGINT, in Spalte 2 den umgesetzten Zeitstempel und in Spalte 3 die 2 Stunden für Sommerzeit drauf addiert und in der letzten Spalte das Datum.

- Berechnung der Sekunden:
Da das Format Millisecond ist, wird bei der Division durch Tausend die Sekunden ermittelt.

```
timestamp ('1970-01-01-00.00.00.000000') + ( when_changed / 1000) SECOND
```

- Berechnung des Datums:
1 Tag besteht aus 24h mit je 60 Minuten und 60 Sekunden, d.h. 86400 Sekunden. Um den Datumswert aus Millisekunden zu ermitteln, muss der Inhalt der BIGINT Spalte durch 86400*1000 Sekunden dividiert werden.

```
date ('01.01.1970') + ( when_changed / 86400000 ) days
```

```
select distinct when_changed
, ( timestamp ('1970-01-01-00.00.00.000000') + ( when_changed / 1000) SECOND ) as calcul_tstamp
, ( timestamp ('1970-01-01') + ( when_changed / 1000) SECOND + 2 HOURS ) as calcul_tstamp
, ( date ('01.01.1970') + ( when_changed / 86400000 ) days) as calcul_date
from DB2NL.CONVERT
fetch first 10 rows only
```

```
WHEN_CHANGED          CALCUL_TIMESTAMP          CALCUL_TIMESTAMP          CALCUL_DATE
-----
1430983423996 2015-05-07-07.23.43.000000 2015-05-07-09.23.43.000000 05/07/2015
1429524708044 2015-04-20-10.11.48.000000 2015-04-20-12.11.48.000000 04/20/2015
1430983427895 2015-05-07-07.23.47.000000 2015-05-07-09.23.47.000000 05/07/2015
1429524755028 2015-04-20-10.12.35.000000 2015-04-20-12.12.35.000000 04/20/2015
1429524895535 2015-04-20-10.14.55.000000 2015-04-20-12.14.55.000000 04/20/2015
1429686400640 2015-04-22-07.06.40.000000 2015-04-22-09.06.40.000000 04/22/2015
1430112906755 2015-04-27-05.35.06.000000 2015-04-27-07.35.06.000000 04/27/2015
1429524895534 2015-04-20-10.14.55.000000 2015-04-20-12.14.55.000000 04/20/2015
1429686400639 2015-04-22-07.06.40.000000 2015-04-22-09.06.40.000000 04/22/2015
1429686400645 2015-04-22-07.06.40.000000 2015-04-22-09.06.40.000000 04/22/2015
```

10 record(s) selected.

Natürlich könnte man das Ganze auch in Excel formatieren. Dabei muss man aber Berücksichtigen, das Exel die Zeit ab 1900 zählt.

1429524708044	ms seit 1970 - Excel rechnet jedoch in Sekunden ab 1900
86400000	Divisor Umrechnung in ms (24h * 60 min * 60s * 1000)
7200000	Sommerzeit (2h * 60min * 60s * 1000)
16545,50819	Anzahl der Tage (inklusive Sommerzeit) (Zeitwert + Sommerzeit / Divisor der Umrechnung)
18.04.1945 12:11:48	Reine Zeit von Excel, mit Zeitrechnung von 1900 beginnend
25569	Anzahl Tage 1900 - 1970
01.01.1970 00:00:00	Zeitwert für Anzahl Tage von 1900-1970
42114,5082	Anzahl der Tage seit 1970 zuzüglich der Anzahl Tage seit 1900
20.04.2015 12:11:48	-Ergebnis als Datum formatiert

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

Eine Liste der anstehenden Konferenzen ist [hier](#) zu finden.

DB2 Regionale Usergroup

Das 8. DeDUG Meeting (DB2 LUW User Group) findet am 3. Juli 2015 in München (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/softwareerviceshop

Dort finden Sie zahlreiche Software Service Angebote, darunter auch die beiden folgenden Informix 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
Andrea Ott	IT-Spezialist Artikel: „Stolpersteine beim Einsatz von DB2 HADR mit ACR und TSA“

Reviewer und Ideenlieferanten:

Dirk Fechner	IBM SWG
Peter Schurr	IBM SWG