

Willkommen zum „IBM DB2 Newsletter“

Liebe Leserinnen und Leser,

Teile von Deutschland sind bereits in den Ferien oder fast wieder daheim.

Die Redaktion des DB2 NL nimmt insofern darauf Rücksicht, das die Sommer-Ausgabe immer noch auf einige Seiten ausgedruckt und dann mit an den Strand, ins Freibad oder auf die Terrasse mitgenommen werden kann.

Genießen Sie Ihren Urlaub und vergessen Sie nicht, die Tipps nach dem Urlaub auch mal auszuprobieren.



Viel Spaß mit den Tipps & Tricks der aktuellen Ausgabe.

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

Ihr TechTeam

Inhaltsverzeichnis

ARTIKELSERIE: IBM OPTIM - ENTERPRISE DATA MANAGEMENT.....	2
TECHTIPP: VERWENDUNG DES DB2 RESTORE BEFEHLS MIT DER REBUILD OPTION.....	4
SZENARIO 1 – REBUILD DER KOMPLETTEN DATENBANK MIT VERSCHIEDENEN BACKUPS.....	5
SZENARIO 2 – PARTIELLES DATENBANK REBUILD EINES TABLESPACES.....	6
TECHTIPP: NEUER DB CFG PARAMETER BLOCKNONLOGGED.....	7
TECHTIPP: ERMITTELN VON SQL STATEMENTS AUS EINEM BIND-FILE.....	8
SCHULUNGEN / TAGUNGEN / INFORMATIONSVERANSTALTUNG.....	9
DB2 AKTUELL 2009.....	9
CHATS MIT DEM LABOR.....	9
NEWSLETTER ARCHIV.....	10
ANMELDUNG/ABMELDUNG.....	10
DIE AUTOREN DIESER AUSGABE:.....	10
A SMARTER PLANET:	10

Artikelserie: IBM Optim - Enterprise Data Management

IBM hat kürzlich eine Vielzahl seiner Data Studio-Produkte unter dem Namen „Optim“ zusammengeführt. Dies bietet die Möglichkeit, eine vollständig integrierte Umgebung zur Datenverwaltung bereitzustellen - eine umfassende Lösung für Entwurf, Entwicklung, Bereitstellung und Verwaltung von Datenbankanwendungen während des gesamten Lebenszyklus von Daten.

Dieser Artikel bezieht sich auf die IBM Optim Enterprise Data Management Lösung (Optim

Data Growth, Optim Test Data Management und Optim Data Privacy), im Folgenden Optim genannt.

Optim ist eine breit gefächerte und heterogene Datenmanagement Lösung, die Unternehmen hilft Probleme wie Datenwachstum, Testdatenmanagement und Datenschutz in den Griff zu bekommen.

Optim bietet eine Lösung Unternehmensdaten durch jedes Stadium des Information Lifecycles zu führen.

Unterstützt werden die Plattformen z/OS, Linux, Unix, Windows sowie alle gängigen DBMs (DB2, Oracle, Sybase, SQL Server, Informix, IMS, VSAM, ...). Außerdem unterstützt Optim die ERP und CRM Schlüsselapplikationen (Oracle E-Business Suite, PeopleSoft Enterprise, JD Edwards EnterpriseOne, Siebel, Amdocs CRM, ...).

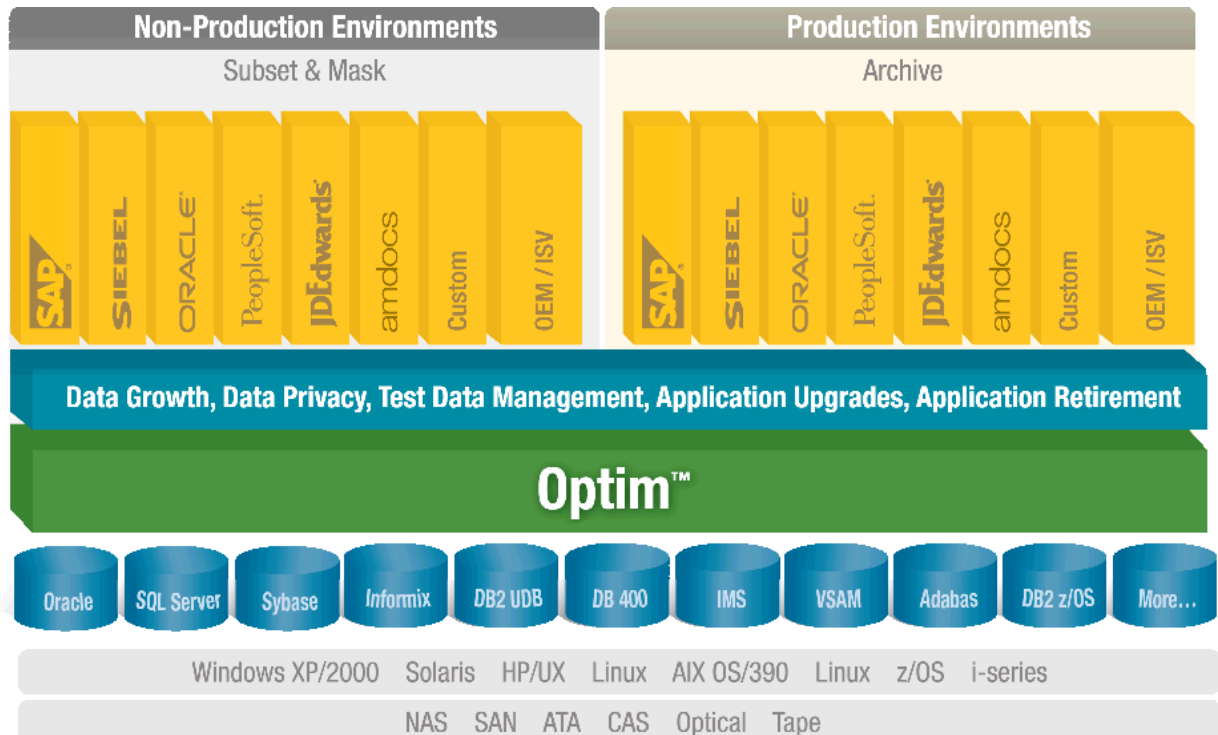


Abbildung 1- Optim: unterstützte Plattformen und DBMS

Die Enterprise Data Managementlösung lässt sich in die Teilbereiche Optim Data Growth, Optim Test Data Management und Optim Data Privacy unterteilen.

- Die **Optim Data Growth Lösung** ermöglicht es schnell wachsende Datenmengen durch (regelmäßige) Archivierung historischer Daten in ein sicheres Archiv in den Griff zu bekommen. Das Archiv kann in einer Datenbank oder als komprimierte Datei (in Optim proprietärem Format, verschlüsselt) gehalten werden und auf Hardware jeglicher Art gespeichert werden. Auf die archivierten Daten kann u.a. mit JDBC zugegriffen werden, so dass den gesetzlichen Regelungen (Daten müssen 10 Jahre zugänglich aufbewahrt werden, ...) Genüge getan wird.

Archivierung historischer oder selten benutzter Daten führt zu einer verbesserten Performance der eingesetzten Anwendungen und geringere Speicherkosten. Archivierung vor einem Anwendungs-Upgrade reduziert die Menge der zu migrierenden Daten beträchtlich und verringert somit auch die benötigte Downtime. Daten von alten, nicht mehr unterstützten Applikationen können mit Optim archiviert werden, die Applikation kann danach stillgelegt werden.

- Die **Optim Test Data Management Lösung** ermöglicht es Firmen neue Applikationen oder Upgrades von Applikationen schneller einzusetzen und somit die Einnahmen zu erhöhen. Durch die Extraktion eines genau definierten Subsets der Anwendungsdaten

kann eine Testdatenbank in der gewünschten Größe schneller und einfacher erstellt werden, als wenn die ganze Produktionsdatenbank geklont werden muss. Mit der geringeren Datenmenge können iterative Testläufe schneller durchlaufen werden. Die Identifikation und Fehlerbehebung in den ersten Phasen der Entwicklung ermöglicht den Testern die Qualität zu erhöhen.

- Die **Optim Data Privacy Lösung** bietet an Kontext und Anwendung angepasste und persistente Maskierungstechniken. Durch die Ersetzung der Wirkbetriebsdaten durch anonymisierte Daten, erstellt Optim eine geschützte Testdatenbank, die trotzdem akkurate und aussagekräftige Testergebnisse garantiert, sowie die Einhaltung von Gesetzen wie HIPAA, GLBA, PCI und der EU Direktive zum Datenschutz.

In diesem Artikel soll die Anonymisierung von in DB2 LUW gehaltenen Daten durch Optim vorgestellt werden.

Datenanonymisierung erlaubt es Entwicklern, Testern und Trainern realistische Daten zu benutzen, die valide Ergebnisse produzieren und dennoch keine sensiblen Daten preisgeben.

Optims Datenmaskierungstechnologie bewahrt die Integrität der Daten und produziert konsistente und akkurate Testergebnisse, die die Applikationslogik widerspiegeln.

Applikationslogik bewahrende Datenmaskierung

Optims Datenmaskierungstechnik verarbeitet die Originaldaten so, dass auch die maskierten Daten die Applikationslogik nicht verletzen. Zum Beispiel werden Nachnamen mit anderen Nachnamen (aus einer Umsetztabelle) ersetzt, nicht mit bedeutungslosen Textstrings. Auch numerische Felder behalten ihre korrekte Struktur bei. Bestehen beispielsweise diagnostische Codes aus vier Ziffern im Wertebereich von 0001 bis 1000, wäre eine Maskierung mit dem Wert 2000 im Kontext des Applikationstests ungültig. Sehr wichtig ist außerdem, dass Optim alle maskierten Datenelemente konsistent in der Testdatenbank propagiert.

Kontext bewahrende Datenmaskierung

Optim bietet eine Vielzahl von Maskierungstechniken für unterschiedliche Arten von sensiblen Informationen wie z.B. Geburtsdaten, Kontonummern, Kreditkartennummern, Email-Adressen. Die Anonymisierung geschieht u.a. mit Hilfe der Optim Transformation Library Routinen und Optim länderspezifischen Umsetzungsstabellen.

Persistente Datenmaskierung

Optims Maskierungstechnik generiert persistente Ersetzungswerte für die Werte der gewünschten Tabellenspalten und propagiert die neuen Werte konsistent über Applikations-, Datenbank-, Betriebssystem- und Plattformebenen hinweg. Dies garantiert die Skalierbarkeit der Anonymisierung.

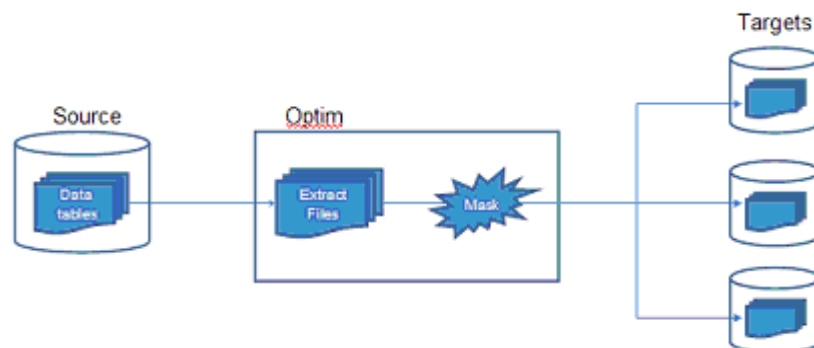


Abbildung 2- Ablauf der Optim Datenmaskierung von der Produktions- zur Testumgebung
Im Folgeartikel wird die praktische Umsetzung von Data Privacy mit Optim beschrieben werden.

Links:

- [Optim Produkt-Familie](#)

- [Optim Lösungen für 'Integriertes Data Management'](#)
- [Optim Enterprise Data Management Solution](#)
- [Optim Integrated Data Management](#)
- [Optim Data Growth](#)
- [Optim Test Data Management](#)
- [Optim Data Privacy](#)

TechTipp: Verwendung des DB2 Restore Befehls mit der Rebuild Option

Mit DB2 Version 9.1 wurde für den Restore- Befehl eine zusätzliche Rebuild- Option hinzugefügt, welche zur erneuten Datenbankerstellung verwendet werden kann.

Für den Restore mit der Rebuild-Option kann ein Datenbank-Backup, ein Tablespace-Backup oder beides, basierend auf einem Full-Database-Backup oder einem Inkrementellen Backup verwendet werden.

Im Recovery-Fall können dringend benötigte Tablespaces schneller wieder hergestellt werden, als z.B andere Tablespaces, die auch zu einem späteren Zeitpunkt eingespielt werden können.

Es bietet sich zudem die Möglichkeit eine Test-Datenbank aus einem Subset der Produktionsdaten herzustellen. Dies ist hilfreich, da sehr große Tabellen oft mit Range-Partitioning angelegt werden, wobei jede Range einen separaten Tablespace benutzen kann. So können z.B. aus große Tabellen mit Inhalten der letzten 3 Jahre, nur der letzte Monat mit der Restore-Rebuild Option auf das Testsystem kopiert werden.

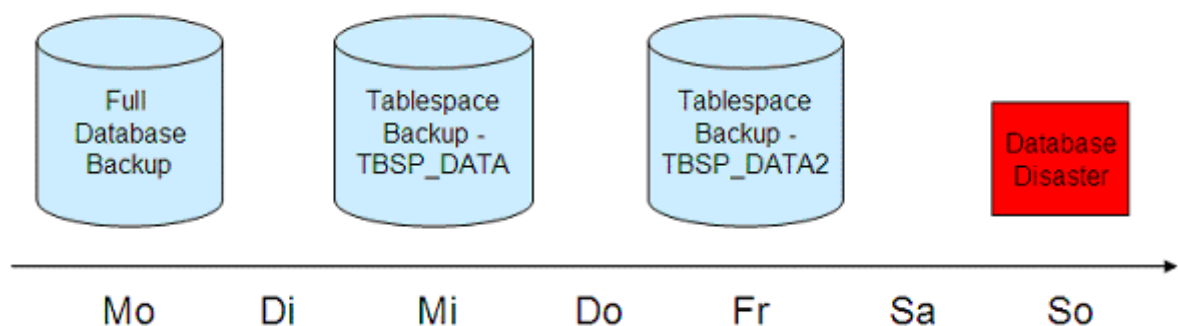
Rebuild Option des Restore Kommandos:

```
>>-RESTORE--++-DATABASE+---source-database-alias----->
>-+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+----->
+-REBUILD WITH-----++-ALL Tablespaces IN DATABASE--+-+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
|                                     | '-ALL Tablespaces IN IMAGE----' '-EXCEPT-|rebuild-tbsp-clause|- ' | |
|                                     | '-| rebuild-Tablespace-clause |-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----'-----
|                                     |
```

Testumgebung:

- DB-Name: SAMPLEDB
- Für Tablespace Backups muss bereits Archival Logging aktiviert sein

Szenario 1 – REBUILD der kompletten Datenbank mit verschiedenen Backups



- Wöchentliche Durchführung von:
 - Full Database Backup am Montag (Offline oder Online)
 - Tablespace- Backup von TBSP_DATA am Mittwoch

- Tablespace- Backup von TBSP_DATA2 am Freitag
- DB-Desaster am Samstag
- Restore mit Rebuild- Option

```
restore database SAMPLEDB REBUILD WITH ALL Tablespaces  
IN DATABASE TAKEN AT <Timestamp von Freitags Tablespace backup >
```

=> Es wird die Recovery-History-Datei nach den aktuellsten Backups für jeden Tablespace durchsucht. Diese werden dann automatisch wiederhergestellt.

- Rebuild kann auch in einem partitionierten Datenbank Umgebung (DPF) verwendet werden
 - Dafür muss der Syscat-Tablespace enthalten sein oder als Erstes in der Katalog Partition wiederhergestellt werden.
 - Bei einer DPF Datenbank muss ein Restore mit der Rebuild Option für jede Datenbank-Partition durchgeführt werden (mit db2_all Option)
- Restore-Befehl Beispiel für DPF Datenbanken:

- Restore mit Rebuild der Katalog Partition von letzten Syscatspace Tablespace Backup.

```
db2_all "<<<+0< db2 -tvf <path>/restore_syscat.sql"
```

Datei: restore_syscat.sql:

```
restore database SAMPLEDB REBUILD WITH  
ALL Tablespaces IN DATABASE  
from '/data $N /backup','/data $N /backup'  
TAKEN AT 20090723162832 WITHOUT PROMPTING;
```

- Restore mit Rebuild der restlichen Datenbank des letzten Tablespace Backups (von einem Tablespace, der auf allen Datenbank Partitionen außer dem Katalog existiert)

```
db2_all "<<<-0< db2 -tvf <path>/restore_Tablespace.sql"
```

Datei: restore_Tablespace.sql:

```
restore database SAMPLEDB REBUILD WITH  
ALL Tablespaces IN DATABASE  
from '/data $N /backup','/data $N /backup'  
TAKEN AT 20090723162836 WITHOUT PROMPTING;
```

- Rollforward

```
rollforward db SAMPLEDB to end of logs and stop
```

=> Aufgrund der verschiedenen Tablespace-Backups, die während der Woche durchgeführt worden sind, werden weniger Log-Files benötigt

- Vor DB2 V9 musste zuerst das Full-Database-Backup wiederhergestellt werden, bevor aktuellere Tablespace-Backups eingespielt werden konnten.
=> Durch die Rebuild-Funktionalität sind weniger Restore-Kommandos notwendig.

Szenario 2 – Partielles Datenbank Rebuild eines Tablespaces

- Tablespace Backups

```
BACKUP DATABASE SAMPLEDB  
Tablespace (SYSCATSPACE, TBSP_DATA, TBSP_INDEX, TBSP_DATA2, TBSP_INDEX2)  
to '/data/backup';
```

- Tablespace Backups Beispiel für DPF Datenbanken:

```
BACKUP DATABASE SAMPLEDB  
ON DBPARTITIONNUM (0)
```

```
tablespace (SYSCATSPACE)
to '/data $N /backup','/data $N /backup';
```

```
BACKUP DATABASE SAMPLEDB
ON ALL DBPARTITIONNUMS EXCEPT DBPARTITIONNUM (0)
tablespace (TBSP_DATA, TBSP_INDEX, TBSP_DATA2, TBSP_INDEX2)
to '/data $N /backup','/data $N /backup';
```

- Drop Database:

```
drop db SAMPLEDB
```

- Restore nur von TBSP_DATA, aber Syscatspace ist notwendig

```
restore database SAMPLEDB REBUILD WITH
Tablespace (SYSCATSPACE, TBSP_DATA)
from '/data/backup'
taken at 20090721104444
WITHOUT PROMPTING;
```

- Restore-Befehl Beispiel für DPF Datenbanken:

- Im DPF- Umfeld muss der Syscatspace zuerst restored werden, da dieser nur auf einer Partition liegt und er daher einzeln als erstes wiederhergestellt werden muss

Restore mit Rebuild der Syscatspace

```
db2_all "<<<+0< db2 -tvf <path>/restore_syscat.sql"
```

Datei: restore_syscat.sql:

```
restore database SAMPLEDB REBUILD WITH
Tablespace (SYSCATSPACE) from '/data $N /backup','/data $N /backup' TAKEN AT
20090723162832 WITHOUT PROMPTING;
```

- Restore mit Rebuild der gewünschten partitionierten Tablespaces

```
db2_all "<<<-0< db2 -tvf <path>/restore_Tablespace.sql"
```

Datei :restore_Tablespace.sql:

```
restore database SAMPLEDB REBUILD WITH
Tablespace (TBSP_DATA, TBSP_INDEX )
from '/data $N /backup','/data $N /backup','/db2fs3p $N /backup','/data $N /backup'
TAKEN AT 20090723162836 WITHOUT PROMPTING;
```

- Rollforward:

```
db2 rollforward db SAMPLEDB to the end of logs and stop
```

- Für DPF- Datenbank erhält man folgende Warnung:

```
db2 rollforward db SAMPLEDB to end of logs and stop
```

```
SQL1271W Database "SAMPLEDB" is recovered but one or more table spaces are
off-line on node(s) "0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16".
```

=> Man kann nur auf Tabellen des einen Tablespace zugreifen

Hinweise:

- Alle abhängigen Tablespaces (z.B. Indexe haben separaten Tablespace) sollen wiederhergestellt werden bevor man die Tabellen benutzen kann
- Alle nicht berücksichtigten Tablespace des partiellen Datenbank Rebuilds werden in den "Restore Pending" Status gesetzt
 - Diese Tablespaces können später wiederhergestellt werden
 - Nicht benötigte Tablespaces können entfernt werden
 - Vor DB2 V9 konnten keine Tablespace- Backups vor dem Restore des Full-Database- Backup eingespielt werden.

- Die Rebuild-Option kann zusammen mit dem Redirected Restore verwendet werden
 - Die Tablespace- Container können angepasst werden
- Point-In-Time Recovery Unterstützung für Tablespace Backups
- Das Monitoring eines Restores mit Rebuild Option ist mit dem list utilities Kommando möglich:

```
db2 list utilities show detail

ID                = 93
Type              = RESTORE
Database Name     = SAMPLEDB
Partition Number  = 0
Description       = automatic db rebuild
Start Time        = 07/23/2009 16:36:21.034442
State             = Executing
Invocation Type   = User

Progress Monitoring:
Completed Work    = 96690176 bytes
Start Time        = 07/23/2009 16:36:21.034446
```

Weitere Informationen sind dazu im [Information Center](#) zu finden.

TechTipp: Neuer DB CFG Parameter BLOCKNONLOGGED.

Mit DB2 UDB LUW V9.5 FP4 wurde ein neuer DB Konfigurationsparameter eingeführt:

Block non logged operations (BLOCKNONLOGGED)

Näheres dazu ist im [Information Center](#) zu finden.

Dieser Parameter ist besonders im HADR Umfeld interessant.

Ein LOAD mit NORECOVERABLE oder ein ALTER TABLE ... ACTIVATE NOT LOGGED INITIALLY auf dem PRIMARY führen dazu, dass die betreffende Tabelle auf dem STANDBY ungültig wird, da es sich um sogenannte non logged Operationen handelt.

Folgende Meldung wird im db2diag.log des STANDBY für die betroffene Tabelle angezeigt:

```
2009-06-10-11.18.26.771506+000 E125331A695          LEVEL: Warning
PID       : 2592986                TID  : 1          PROC : db2redom (SAMPLEDB) 0
INSTANCE: db2hadr                 NODE : 000        DB   : SAMPLEDB
APPHDL   : 0-86                   APPID: *LOCAL.DB2.090610111317
FUNCTION: DB2 UDB, data management, sqlMarkObjInErr, probe:1
MESSAGE  : ADM5571W DB2 is marking an object "DATA" with id "xx" in Tablespace
           "yy" for table "TBSPACEID=yy.TABLEID=xx" unavailable.  If the object is
           a table it will have to be dropped.  If the object is a partition it
           will have to be detached.  If the object is a non-partitioned index
           the index will have to be dropped.
```

Um non logged Operationen wie ALTER TABLE ... ACTIVATE NOT LOGGED INITIALLY oder LOAD mit NORECOVERABLE am Primary zu verhindern, kann BLOCKNONLOGGED in der DB CFG auf YES gesetzt werden.

Dann erhält der User folgende Fehlermeldungen:

a)

```
$ db2 "alter table t1 activate not logged initially"
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0628N Multiple or conflicting keywords involving the "ACTIVATE NOT LOGGED
INITIALLY" clause are present. SQLSTATE=42613
```

b)

```
$ db2 "load from t.unl of del insert into t1 nonrecoverable"
SQL2032N The "COPY FLAG" parameter is not valid. SQLSTATE=22531
```

Somit kann man am Primary verhindern, dass solche non logged Operationen ausgeführt werden und die Daten auf den Standby invalidiert werden.

Der Parameter ist zwar hauptsächlich für HADR sinnvoll, kann aber auch ohne HADR genutzt

werden.

TechTipp: Ermitteln von SQL Statements aus einem Bind-File

Immer wieder kommt mal die Frage auf, was steckt eigentlich in einem *.bnd File.

Mit folgendem Kommando kann das ermittelt werden.

```
db2bfd -b -s <*.bnd-File>
```

Beispiel

```
db2bfd -b -s ./tpcd/driver/tpcdUF.bnd
```

```
./tpcd/driver/tpcdUF.bnd: Header Contents
```

Header Fields:

Field	Value
releaseNum	0x800
Endian	0x42
numHvars	5
maxSect	4
numStmt	21
optInternalCnt	4
optCount	11

Name	Value
Isolation Level	Read Stability
Creator	"PERFPOL1"
App Name	"TPCDUF "
Timestamp	"MBZPAeEX:2007/04/30 00:15:25:74"
Cnulreqd	Yes
Sql Error	No package
Block	Block All
Date	ISO
Time	ISO
Degree	1
Validate	Bind

*** All other options are using default settings as specified by the server ***

```
./tpcd/driver/tpcdUF.bnd: SQL Statements = 21
```

Line	Sec	Typ	Var	Len	SQL statement text
90	0	5	0	21	BEGIN DECLARE SECTION
96	0	2	0	19	END DECLARE SECTION
182	0	19	1	21	CONNECT TO :H00001
184	0	19	3	54	CONNECT TO :H00001 USER :H00002 USING :H00003
217	1	0	1	269	INSERT INTO TPCD.ORDERS SELECT O_ORDERKEY, O_CUSTKEY, O_ORDERSTATUS, O_TOTALPRICE, O_ORDERDATE, O_ORDERPRIORITY, O_CLERK, O_SHIPPRIORITY, O_COMMENT FROM TPCDTEMP.ORDERS_NEW WHERE APP_ID = :H00004
254	2	0	1	350	INSERT INTO TPCD.LINEITEM SELECT L_ORDERKEY, L_PARTKEY, L_SUPPKEY, L_LINENUMBER, L_QUANTITY, L_EXTENDEDPRICE, L_DISCOUNT, L_TAX, L_RETURNFLAG, L_LINESTATUS, L_SHIPDATE, L_COMMITDATE, L_RECEIPTDATE, L_SHIPINSTRUCT, L_SHIPMODE, L_COMMENT FROM TPCDTEMP.LINEITEM_NEW WHERE APP_ID = :H00004
295	0	8	0	11	COMMIT WORK
339	0	13	0	13	ROLLBACK WORK
346	0	19	0	13	CONNECT RESET
484	0	19	1	21	CONNECT TO :H00001
486	0	19	3	54	CONNECT TO :H00001 USER :H00002 USING :H00003
518	3	0	1	174	DELETE FROM TPCD.LINEITEM WHERE L_ORDERKEY IN (SELECT O_ORDERKEY FROM TPCDTEMP


```
MP.ORDERS_DEL                                WHERE APP_ID = :H00004
)
560  4  0  1 139 DELETE FROM TPCD.ORDERS        WHERE O_ORDERKEY IN
                                     (SELECT O_ORDERKEY FROM TPCDTEMP.ORDERS_DEL
                                     WHERE APP_ID = :H00004 )
605  0  8  0  11 COMMIT WORK
643  0 13  0  13 ROLLBACK WORK
648  0 19  0  13 CONNECT RESET
839  0 19  1  35 CONNECT TO :H00001          IN SHARE MODE
853  0 19  0  13 CONNECT RESET
917  0 19  1  21 CONNECT TO :H00001
1147 0  8  0  6 COMMIT
1167 0 19  0  13 CONNECT RESET-
```

Weiterführende Information zu diesem Befehl sind im [Information Center](#) zu finden.

Schulungen / Tagungen / Informationsveranstaltung

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

DB2 Aktuell 2009

DB2 Aktuell 2009
15.-16. September
in Münster



Die vorläufige Agenda für diese technische Konferenz steht [hier](#) zum Download zur Verfügung.

Anmeldungs- und andere Veranstaltungsinformationen erhalten sie [hier](#).

Chats mit dem Labor

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

Die Präsentationen der Chats, können angeschaut und heruntergeladen werden.

Der letzte DB2 Chat am 16.07 war zum Thema „Enabling applications from Oracle to DB2 the easy way“

Analog zu den DB2 Chats gibt es auch noch BI Chats mit dem Labor. Die Präsentationen und Replays der Chats können [hier](#) angeschaut und heruntergeladen werden.

Der letzte BI Chat am 17.07 „ The Journey to Trusted Information - Workload Management in DB2 9.7“ sollte bald zum Herunterladen verfügbar sein.

Newsletter Archiv

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

- Lis.Tec: http://www.listec.de/DB2_Newsletter/View_category.html

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 Subjekt „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	IT-Spezialist für DB2 LUW, IBM SWG; djs@de.ibm.com
Sascha Laudien	IBM SWG, IT Spezialist
Annetta Fourkiotis	Data Specialist for DB2 LUW and Optim, IBM SWG; Annetta.Fourkiotis@de.ibm.com

Reviewer und Ideenlieferanten:

Nela Krawez	IBM SWG, InfoSphere Balanced Warehouse Development
Wilfried Hoge	IBM SWG, Technical Sales IM

A Smarter Planet:

IBM SWG IM Services in enger Zusammenarbeit mit Business Partnern und ISV's.

SWG Information Management Services bietet seit Anfang des Jahres IBM Business Partnern, Distributoren und ISV's eine enge Zusammenarbeit in allen Bereichen rund um die IBM Information Management Produkte an.

Im Rahmen der "A Smarter Planet" Initiative soll diese Zusammenarbeit unseren Partnern die Möglichkeit geben neue Themen zu entdecken und Lösungen zu implementieren, welche über den bisherigen Standard hinausgehen: um unsere Welt ein bisschen intelligenter und transparenter zu machen.

Durch unser perfekt aufgestelltes Team von über 120 Personen, in den Bereichen Technik, Architektur und Projektleitung, können wir unseren Partnern genau die Skills und Erfahrungen zur Seite stellen, die es ihnen ermöglichen die neuen Wege zu beschreiten.

Möchten auch Sie mithelfen unsere Welt "smarter" zu machen?
Dann schreiben Sie eine kurze Email an volker.fraenkle@de.ibm.com.