Willkommen zum „IBM DB2 Newsletter“

Liebe Leserinnen und Leser,

Nun die Ferien größtenteils vorbei sind, werden Sie sicherlich Zeit finden, diesen und auch die letzte Ausgabe (05/2009) durchzulesen und ein paar Sachen auszuprobieren.
Ich hoffe Sie hatten einen schönen und erholsamen Urlaub. Das Wetter hat zumindestens gepasst.

Im letzten Newsletter hatten wir den Autor eines Artikels unterschlagen, dies möchten wir entschuldigen und reichen es in dieser Ausgabe nach.

Weiterhin haben wir - wie immer - versucht interessante Beiträge für Sie zusammenzustellen.

Also viel Spaß mit den Tipps & Tricks der aktuellen Ausgabe.

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

Ihr TechTeam

Inhaltsverzeichnis

Nachtrag zur letzten Ausgabe

In der letzten Ausgabe war nur noch ein Archivar zu finden. Dies ist nicht aufgrund einer Bevorzugung entstanden, sondern daher, das beim Review die Links zu den anderen Archivaren nicht mehr funktionierten und wir nur funktionierende Links weitergeben wollten.

Das Problem mit dem im Dokument enthaltenen bytec-Link konnte inzwischen beseitigt werden. Es war kein Problem bei der Firma bytec, sondern ein Problem innerhalb dieses Dokumentes. Die Ursachen wurden in Zusammenarbeit mit Kollegen der Fa. bytec analysiert und der Link im Dokument bereinigt. Vielen Dank an die Fa. bytec die maßgeblich zur Lösung beigetragen hat.

Weiterhin hatten wir in der letzten Ausgabe den Autor des Artikels „NEUER DB CFG PARAMETER BLOCKNONLOGGED" unterschlagen. Dies möchten wir entschuldigen. Dieser Artikel wurde durch Martina Lang vom DB2 UDB LUW Advanced Support der IBM SWG eingereicht.

Artikelserie: IBM Optim - Enterprise Data Management – Teil 2

- Die ***Optim Data Privacy Lösung*** bietet, an Kontext und Anwendung angepasste und persistente Maskierungstechniken. Durch die Ersetzung der Wirkbetriebsdatren durch anonymisierte Daten, erstellt Optim eine geschützte Testdatenbank, die trotzdem akurate 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 dargestellt 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 oder Email-Adressen. Die Anonymisierung geschieht u.a. mit Hilfe der Optim Transformation Library Routinen und Optim-länderspezifischen Umsetzungstabellen.

Persistente Datenmaskierung

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

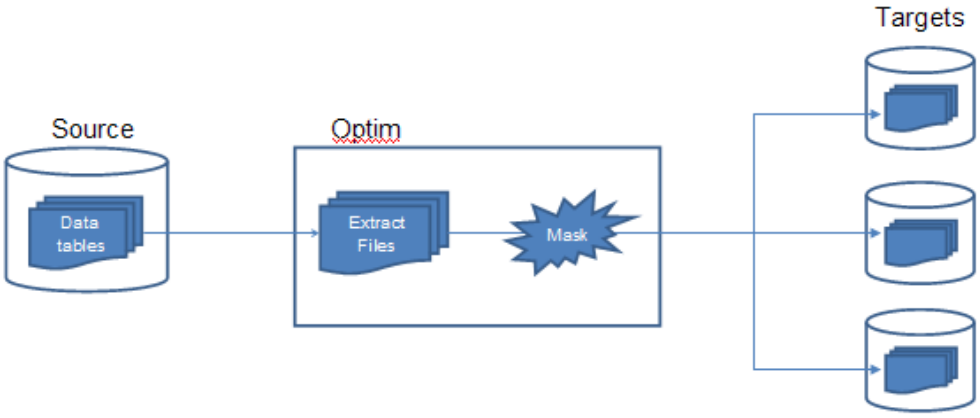


Abbildung 1- Ablauf der Optim Datenmaskierung von der Produktions- zur Testumgebung

Beispiel: Optim Datenanonymisierung am Beispiel von DB2 LUW

Im Folgenden wird beschrieben wie man in DB2 gehaltene Daten mit Optim anonymisiert. Die Installation und Konfiguration von Optim ist sehr einfach, weshalb sie in diesem Artikel nicht erläutert wird.
Als Beispiel-Datenbank wird uns die SAMPLE-DB von DB2 dienen. Es soll die Spalte *Lastname* der Tabelle EMPLOYEE mit einem zufälligen Nachnamen verschlüsselt werden, sowie die Spalte *Location* der Tabelle DEPARTMENT mit einem festen String. Vor der Anonymisierung sollte ein Backup der Datenbank gemacht werden.

Als ersten Schritt legt man eine Datenbank für das Optim Repository an. Das Optim Repository (auch Optim Directory) besteht aus 14 Tabellen. Im Optim Directory sind u.a. Datenbankinformationen hinterlegt (z.B. Verbindungsinformation für SAMPLE DB), außerdem werden dort die vom Nutzer erstellten Tabellenbeziehungen, sowie die Umsetzungstabellen und erstellten Optim Objekte abgespeichert .
Die Datenbank muss mit dem gleichen kodierten Zeichensatz wie die Datenbank angelegt werden, die die zu anonymisierenden Daten enthält (hier UTF-8). Im Beispiel legen wir eine Datenbank mit dem Namen OPTREP wie folgt an:

```
db2 create db OPTREP
```

Die Anonymisierung von Daten mit Optim besteht folgenden Schritten.

1. Zuerst muss eine **Access Defintion** erstellt werden.

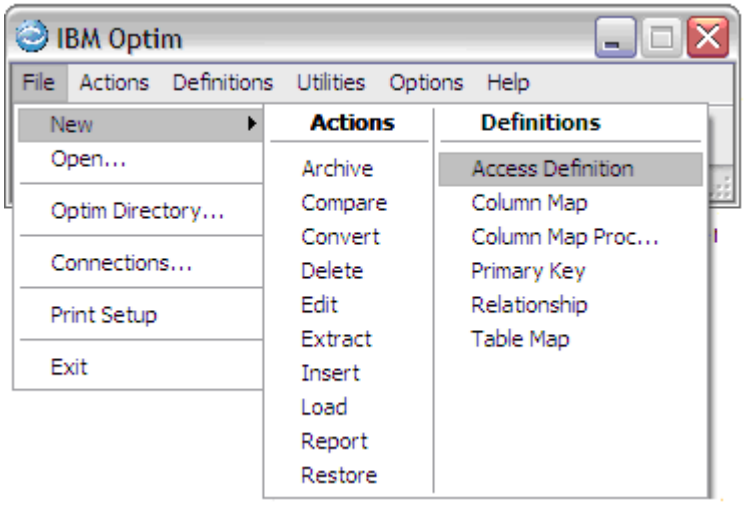


Abbildung 2 - Menue zum Erstellen einer Access Definition

Eine **Access Definition** dient dazu die zu anonymisierenden Tabellen, sowie deren Beziehungen zueinander als referentiell intakte Daten zu definieren. Für dieses Beispiel

definieren wir die Tabellen EMPLOYEE und DEPARTMENT als Starttabellen. Diese Tabellen enthalten die zu anonymisierenden Daten. Von der Tabelle EMPLOYEE soll die Spalte *LASTNAME* und von der Tabelle DEPARTMENT die Spalte *LOCATION* verschlüsselt werden.

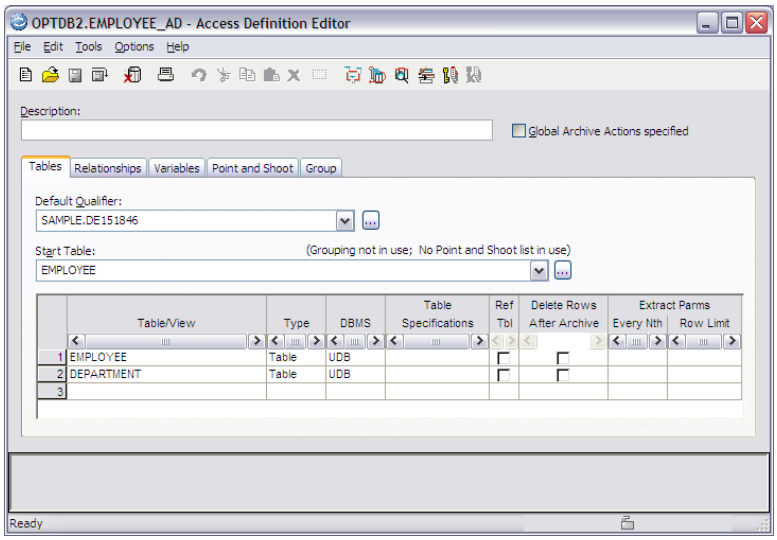


Abbildung 3 - Access Definition: Definition der zu betrachtenden Tabellen

2. Nun möchten wir die Daten dieser Tabellen mit Hilfe eines Optim **Extract Requests** extrahieren. Der Extract Prozess kopiert eine Gruppe zusammenhängender Rows einer oder mehrerer Tabellen und speichert diese in einer Extract-Datei ab. Die anhand der Access Definition extrahierten Daten werden vom Extractprozess weder gelöscht noch modifiziert.

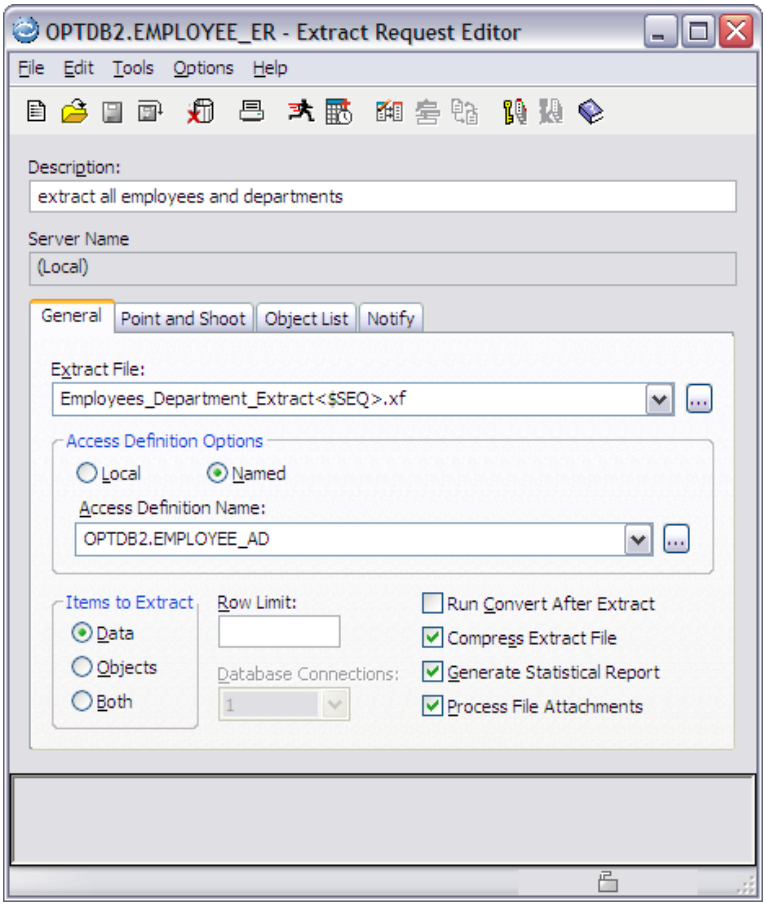


Abbildung 4 - Extract Request: extrahiert die Daten der Tabellen EMPLOYEE und DEPARTMENT

Nach dem Ende des Extract Request wird ein Statusreport angezeigt. Dieser enthält neben allgemeinen Angaben (fehlerloser Lauf bzw. aufgetretene Probleme) auch Informationen zu folgenden Bereichen:

- Wieviel Prozent der Zeit wurde im DBMS verbraucht?
- Wurden Indizes verwendet?
- Wie lange dauerte der Extract?
- Wieviele Rows (pro Tabelle) wurden extrahiert?
- DB2 Lookup Cost
- DB2 Scan Cost
- ...

Anschließend kann der Extract Request entsprechend getuned werden, um die Laufzeit des Extract Requests zu verbessern.

3. Nach der Extrahierung der Daten wird eine **Table Map** definiert. Mit der Table Map legt Optim fest, wie die Tabellen der Extract-Datei auf die Tabellen der Zieldatenbank gemappt werden. Die Tabellennamen können gleich oder unterschiedlich sein.

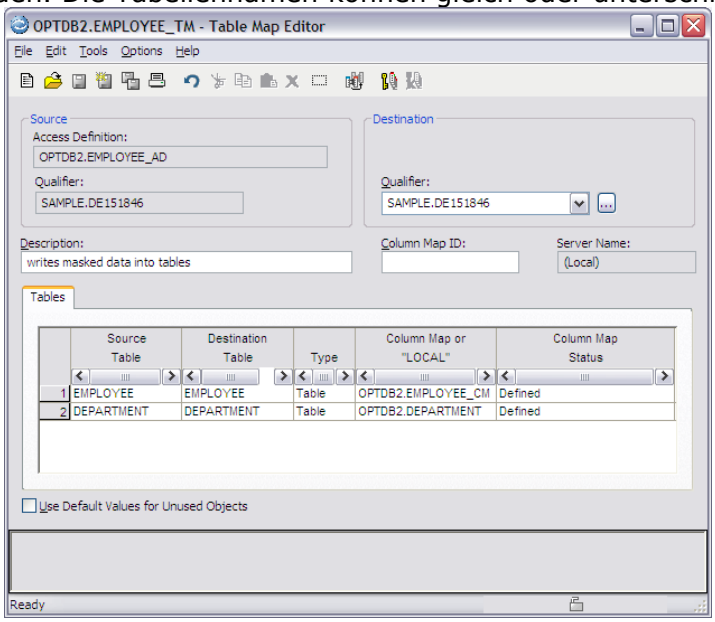


Abbildung 5 - Table Map zur Definition der Zieltabellen

In der Table Map definiert man nun für jede der zu anonymisierenden Tabelle eine **Column Map**. Dabei wird festgelegt, wie die Tabellenspalten der Extract-Datei auf die Tabellenspalten der Zieldatenbank gemappt werden. Wie auch schon bei den Tabellennamen an sich, können die Spaltennamen ebenfalls gleich oder unterschiedlich sein.
Optim bietet für die Anonymisierung u.a. folgenden Maskierungstechniken:

- Ersetzung mit String (z.B. ‚Anonymisiertes Feld')
- Character Substrings
- Zufalls- oder Sequenzzahlen
- Shuffle
- Funktionen, um Emails, Kredikartennummern und Social Security Nummern zu anonymisieren
- Arithmetische Ausdrücke
- Konkatenierte Ausdrücke
- Datumsfunktion (z.B. alle Daten sollen um ein Jahr und 2 Monate zurückgesetzt werden)
- Einsatz von Umsetzungstabellen

Für die Tabelle EMPLOYEE verwenden wir einen Hashlookup und die Optim Umsetztabelle OPTIM_US_LASTNAME (siehe Abbildung 6):

```
HASH_LOOKUP(LASTNAME, SAMPLE.OPTIM.OPTIM_US_LASTNAME(seq, LASTNAME))
```

Hierbei berechnet Optim einen Hashwert für den Wert der Spalte *LASTNAME*, sucht nach diesem Wert in der Umsetztabelle `OPTIM_US_LASTNAME` und ersetzt den alten Nachnamen mit dem entsprechenden neuen Nachnamen aus der Umsetztabelle.

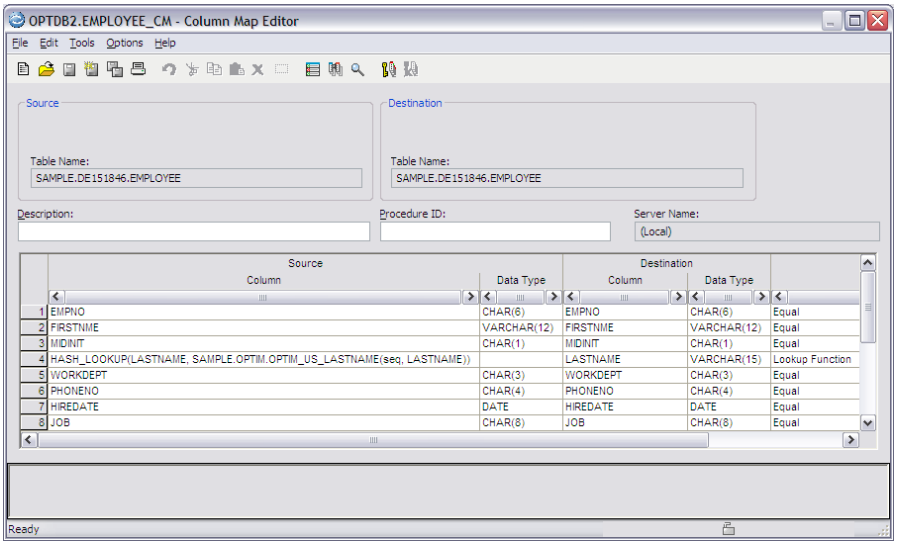


Abbildung 6 - Column Map der Tabelle EMPLOYEE

Für die Tabelle `DEPARTMENT` wird der Wert der Spalte *LOCATION* konsistent mit dem Wert ‚Musterlokation' überschrieben (siehe Abbildung 7).

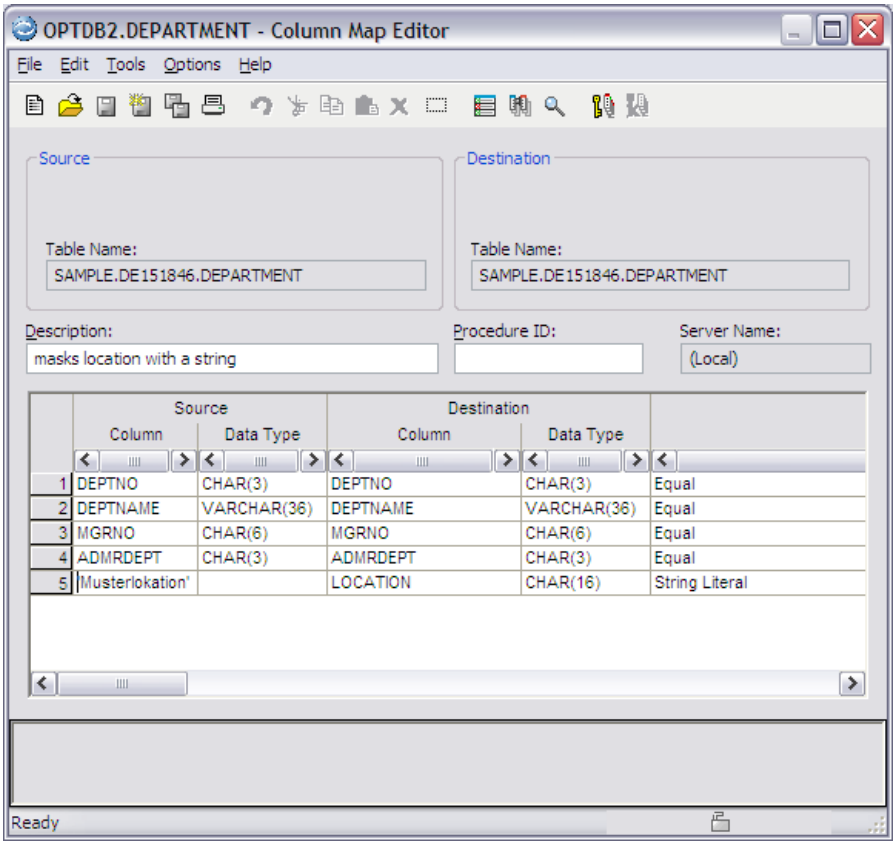


Abbildung 7 - Column Map der Tabelle DEPARTMENT

4. Nun muss die Anonymisierung noch ausgeführt werden und die anonymisierten Daten in die Tabellen EMPLOYEE und DEPARTMENT geschrieben werden. Dies geschieht mit einem **INSERT Request**.
   Der INSERT-Prozess anonymisiert die Daten der Extract-Datei anhand der in der jeweiligen Column Map definierten Maskierungsregeln und fügt anschließend die maskierten Daten in die Zieltabellen (wie in der Table Map definiert) ein.

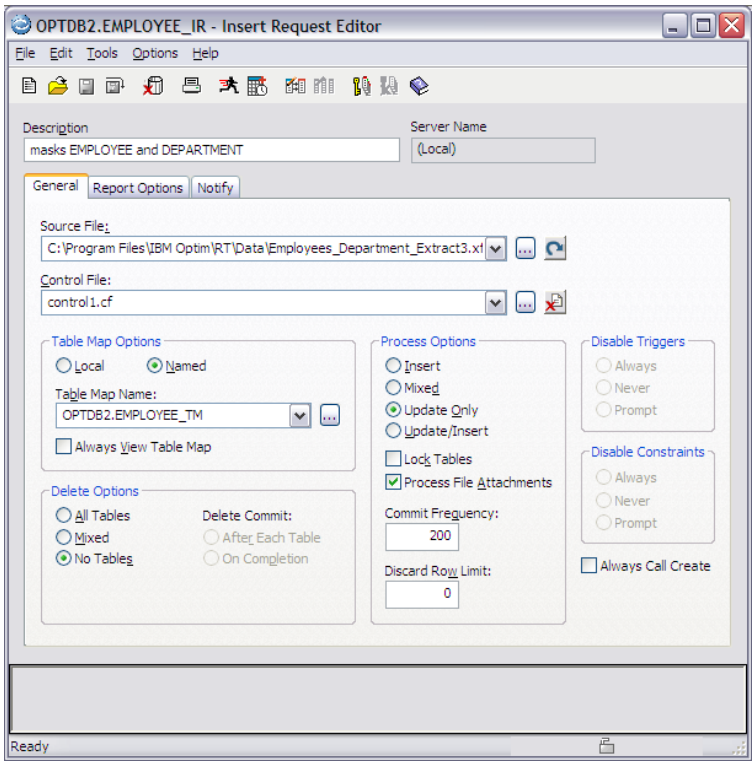
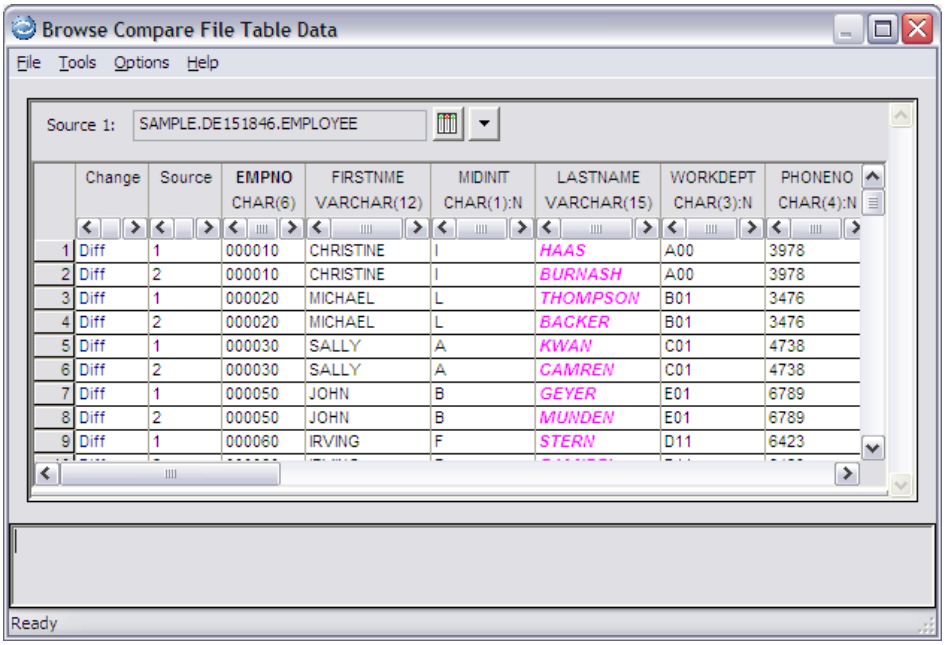


Abbildung 8 - Insert Request führt Anonymisierung aus

Anschließend kann man sich mit dem Optim Browser davon überzeugen, dass die Daten korrekt anonymisiert wurden. Abbildung 9 zeigt, dass die Daten der Spalte Lastname der Tabelle EMPLOYEE erfolgreich anonymisiert wurden. Die unterschiedlichen Werte der originalen und der anonymisierten Daten werden in Magenta angezeigt.



	Change	Source	EMPNO CHAR(6)	FIRSTNME VARCHAR(12)	MIDINIT CHAR(1):N	LASTNAME VARCHAR(15)	WORKDEPT CHAR(3):N	PHONENO CHAR(4):N
1	Diff	1	000010	CHRISTINE	I	*HAAS*	A00	3978
2	Diff	2	000010	CHRISTINE	I	*BURNASH*	A00	3978
3	Diff	1	000020	MICHAEL	L	*THOMPSON*	B01	3476
4	Diff	2	000020	MICHAEL	L	*BACKER*	B01	3476
5	Diff	1	000030	SALLY	A	*KWAN*	C01	4738
6	Diff	2	000030	SALLY	A	*CAMREN*	C01	4738
7	Diff	1	000050	JOHN	B	*GEYER*	E01	6789
8	Diff	2	000050	JOHN	B	*MUNDEN*	E01	6789
9	Diff	1	000060	IRVING	F	*STERN*	D11	6423

Abbildung 9 - Optim Compare von original und anonymisierten Daten der Tabelle EMPLOYEE

Und hier noch mal die Links zur weiterführenden Dokumentation:

- 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: Trusted Connections: Endanwender mit der Userid des Applikationsservers müssen nicht sein

Welcher DBA kennt das Problem nicht ?
Bei den heute üblichen 3-tier Architekturen verbinden sich die Anwender zu einer Middleware und werden dort authentifiziert. Dabei handelt es sich in aller Regel um einen Applikationsserver. Im folgenden wird der Einfachheit halber angenommen, dass es sich um den Websphere Application Server (WAS) handelt. Das eigentliche SQL führen sie dann aber über eine Verbindung des WAS auf die Datenbank durch. Dafür wird in der Regel eine gemeinsame Userid (im folgenden „was_id" genannt) verwendet.
Die Sicherheitsrichtlinien sind prinzipiell erfüllt, sofern das Passwort dieser gemeinsamen Userid „was_id" geheim gehalten wird, da sich der Endbenutzer ja ordnungsgemäß mit seiner eigenen Userid und Passwort beim Applikationsserver authentifiziert hat.
Dieses gängige Verfahren hat jedoch insbesondere für den Datenbankadministrator eine Reihe von gravierenden Nachteilen:

- Auf dem Datenbankserver ist nicht ersichtlich von welchem Enduser das eigentliche Statement kommt. Das ist nicht nur für das Monitoring von Nachteil, sondern auch der Nutzen von weitergehenden Verfahren (Auditing, Workload Management, etc.) wird stark eingeschränkt.
- Der Funktionsuser bekommt die Vereinigungsmenge aller benötigten Rechte der Endbenutzer. Damit besitzt er eine unbeabsichtigte Superuser-Rolle und stellt ein potentielles Sicherheitsrisiko dar.
- Es kann auf Datenbankebene keine granulare Vergabe oder Entzug von Rechten an die einzelnen Benutzer erfolgen.

Ab der DB2 für LUW Version 9.5 bzw. DB2 V9.1 für z/OS gibt es für diese Probleme das Konzept der „Trusted Connection". Der Zugriff auf die Datenbank erfolgt weiterhin über die vom Benutzer „was_id" geöffnete Verbindung, allerdings wird diesmal intern auf die eigentliche Userid des Endbenutzers gewechselt (user-switch).
Die Authentifizierung des Endbenutzers findet wie oben im WAS statt, deshalb verzichtet DB2 nun auf eine erneute Überprüfung des Passworts (und akzeptiert den user-switch). Man traut also der vom Benutzer „was_id" geöffneten Verbindungen, sofern ganz bestimmte Kriterien erfüllt sind.
Um dieses auf Datenbankseite zu definieren, muss ein sogenannter „Trusted Context" definiert werden. Dies geschieht beispielsweise durch folgendes Statement, für das SECADM Berechtigung notwendig ist:

```
CREATE TRUSTED CONTECT mycontext1
BASED UPON CONNECTION USING SYSTEM AUTHID was_id
ATTRIBUTES (   PROTOCOL 'TCPIP',
                        ADDRESS '192.12.45.207',
                        ENCRYPTION 'HIGH' )
ENABLE
ALLOW USER user1, user2, user3;
```

Es entsteht eine „Trusted Connection", wenn alle Attribute des „Trusted Context" <mycontext1> erfüllt sind.
In diesem Fall also beispielsweise wenn der Benutzer „was_id" von der Maschine mit der IP-Adresse 192.12.45.207 eine Verbindung eröffnet und dann als „user2" (user-switch) auf Objekte im DB2 zugreift. Allerdings muss in unserem Beispiel die Verbindung noch eine SSL-Verschlüsselung verwenden, da wir das Attribut ENCRYPTION ‚high' angegeben haben. Andernfalls wären nicht alle Attribute erfüllt und es käme keine „Trusted Connection" zustande.
Informationen über die bestehenden „Trusted Context"-Definitionen sind in den folgenden DB2 Systemtabellen zu finden:

```
SYSCAT.CONTEXTS
SYSCAT.CONTEXTATTRIBUTES
SYSCAT.SURROGATEAUTHIDS
```

Eine „SYSTEM AUTHID" darf nur in einem „Trusted Context" vorkommen. In unserem Beispiel darf also „was_id" nicht noch einmal für einen andere „Trusted Context"-Definition verwendet werden.
Um den Aufwand der Verwaltung der „trusted user" klein zu halten, kann bei der Definition sinnvollerweise „ALLOW USE FOR PUBLIC" verwendet werden. Damit muss der „TRUSTED CONTEXT" nicht bei jedem hinzugekommenen Endanwender angepasst werden.

Am Applikationsserver ist auch eine entsprechende Einstellung vorzunehmen. Beim WAS ist beispielsweise das folgende Database Property zu setzen:

```
propagateClientIdentityUsingTrustedContext
```

Das „TRUSTED CONTEXT"-Konzept erlaubt darüber hinaus nicht nur den Switch einer Userid (hier: was_id -> user2), sondern zusätzlich noch das Annehmen von Rollen.
Also beispielsweise:

```
CREATE TRUSTED CONTECT mycontext2
BASED UPON CONNECTION USING SYSTEM AUTHID report_id
ATTRIBUTES (   PROTOCOL 'TCPIP',
                        ADDRESS 'hostname1.de.ibm.com',
                        ADDRESS 'hostname2.de.ibm.com'
                        ENCRYPTION 'NONE' )
DEFAULT ROLE POWERUSER;
```

D.h. immer wenn der Benutzer „report_id" sich über die Maschinen „hostname1" oder „hostname2" verbindet, bekommt er die POWERUSER-Rolle zugewiesen.
Damit kann der Benutzer „report_id" selbst nur minimale Rechte besitzen und bekommt seine umfassenden Berechtigungen über den passenden „TRUSTED CONTEXT" zugewiesen.
Die Rolle muss natürlich zuvor definiert worden sein, z.B.:

```
CREATE ROLE poweruser;
GRANT DBADM ON DATABASE TO ROLE poweruser;
```

Alternativ hätte Benutzer "report_id" die zuvor definierte Rolle folgendermaßen einnehmen können:

```
SET ROLE poweruser;
```

Dazu wäre es allerdings notwendig gewesen, dass ihm das Recht die Rolle einzunehmen vorher explizit gegeben worden wäre (z.B. mit SECADM Berechtigung):

```
GRANT ROLE poweruser TO USER report_id;
```

Das Rollenkonzept wurde ebenfalls in Version DB2 für LUW V9.5 bzw. DB2 für z/OS V9.1 eingeführt. Es entspricht im Wesentlichen dem im Informix Dynamic Server (IDS) bekannten analogen Feature.

Zusammenfassend lässt sich sagen, dass mit dem „TRUSTED CONTEXT"-Konzept nun eine einfache Möglichkeit besteht, die wirklichen Endbenutzer auch auf der Datenbank zu „sehen". Damit werden die oben angeführten Probleme allesamt behoben.
Der Aufwand die „TRUSTED CONTEXT"-Definitionen einzurichten (und zu verwalten) ist relativ gering.

Ein weiterführender Artikel ist hier zu finden.

TechTipp: Name der Instanz

Sie sind nicht als Instanz Owner angemeldet und wollen Wissen, wie ihre Instanz heißt?
Mit dem Befehlt db2 get instance können Sie es herausfinden. Diese Kommando funktioniert sowohl im Unix, als auch im Windows-Umfeld.

```
> db2 get instance

 The current database manager instance is:  DB2
```

Weiterhin kann jedoch auch die Variable DB2INSTANCE, (die im Unix-Umfeld im db2profile

gesetzt wird) abgefragt werden.

- UNIX

```
dbuser@system /home/dbuser > echo $DB2INSTANCE
db2inst1
```

  Grundlegende Voraussetzung ist, das das db2profile der Instanz geladen wurde. Zum Wechseln der Instanz wird einfach das db2profile der anderen Instanz geladen (z.b. . ~/db2inst2/sqllib/db2profile).

- Windows:

```
C:\Documents and Settings\Administrator>echo %DB2INSTANCE%
DB2_01
```

  Zum Wechseln der Instanz kann bei Windows einfach die DB2INSTANCE Variable neu definiert werden (z.B. set DB2INSTANCE=DB2_02), Voraussetzung dafür ist jedoch, das die Instanz auch existiert.

TechTipp: Ermitteln der Pfade für Datenbankobjekte

Es gibt immer wieder Views, die seit DB2 Version 9 eingeführt wurden, jedoch u.U. immer

noch nicht verwendet werden und die Abfrage mitunter auf die bisherige Art und Weise ausgeführt werden. Hier ein Beispiel dazu, welche Informationen der Administation View DBPATHS zurückliefert.

```
select substr(type,1,30) as db_path_type, substr(path,1,50) as path_name
  from sysibmadm.dbpaths
  order by 1
DB_PATH_TYPE                   PATH_NAME
------------------------------ --------------------------------------------
DBPATH                         /database/system/db2inst1/SAMPLE/NODE0000/SQL00001
DB_STORAGE_PATH                /database/system/db2inst1/SAMPLE/fs01
DB_STORAGE_PATH                /database/system/db2inst1/SAMPLE/fs02
LOCAL_DB_DIRECTORY             /database/system/db2inst1/SAMPLE/NODE0000/sqldbdir/
LOGPATH                        /database/logs/db2inst1/SAMPLE/NODE0000/
TBSP_CONTAINER                 /database/data/db2inst1/TSP_DMS/TABLE1
TBSP_CONTAINER                 /database/data/db2inst1/TSP_DMS/TABLE2
TBSP_DIRECTORY                 /database/data/db2inst1/TSP_SMS2/
TBSP_DIRECTORY                 /database/data/db2inst1/TSP_SMS1/
```

TechTipp: DB2 V9.7 download möglich

Auf der Seite http://www.channeldb2.com/page/download-now wird der Download der DB2 Version 9.7 angeboten.

DB2 Express-C 9.7
Die freie Version des DB2 Servers kann hier in Abhängigkeit der ausgewählten Platform hier runtergeladen werden:
Windows 32-bit | Linux 32-bit | Mac OS X | Windows 64-bit | Linux 64-bit | Linux on POWER | Solaris x64

DB2 Enterprise 9.7
Eine 90 tägige DB2 9.7 Testversion kann hier heruntergeladen werden.

Schulungen / Tagungen / Informationsveranstaltung

Eine Liste der anstehenden Konferenzen ist hier zu finden.

DB2 Aktuell 2009

Die Tagung kommt immer näher. Sie sind noch nicht angemeldet oder wissen nicht, was auf der Agenda steht?

Anmeldungs- und andere Veranstaltungsinformationen erhalten sie hier.

Chats mit dem Labor

Das Auftreten des Chat mit dem Lab hat sich verändert und hat eine direkte Verbindung zum db2 channel bekommen, um vorherige Chats zu wiederholen.

Eine Liste der bereits durchgeführten Chats ist hier zu finden.
Die Präsentationen der Chats, können angeschaut und heruntergeladen werden.

Der letzte DB2 Chat am 27.08 war zum Thema „Simplifying DB2 Administration and Development with IBM Data Studio "

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.

Newsletter Archiv

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

- Lis.Tec
- Cursor Software AG: die alte Adresse (enthalten bis in 200904) ist nicht mehr gültig. Hier der neue Zugang.
- Bytec

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
Jürgen Buck	IBM SWG, IT Spezialist, Informix, DB2 Artikel: Trusted Connections
Annetta Fourkiotis	Data Specialist for DB2 LUW and Optim, IBM SWG; Artikel: IBM Optim - Enterprise Data Management Annetta.Fourkiotis@de.ibm.com
Jens Bobe	IBM SWG, IT Spezialist Artikel: DB2 V9.7 download möglich

Reviewer und Ideenlieferanten:

Nela Krawez	IBM SWG, InfoSphere Balanced Warehouse Development
Frank Berghofer	IT-Spezialist für DB2 LUW, IBM SWG

Andrea Ott	IBM SWG, IT-Spezialist
Gerhard Müller	IBM SWG, IT-Spezialist

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.