

Testdatenanonymisierung mit dem Oracle Data Mask Pack

Bernhard Eichhorn
MIC Management Consulting GmbH
München

MIC Management Consulting GmbH
Kronacher Str. 4 81549 München
Tel. +49 / 89 / 680 711 61
www.mic-muenchen.de
office@mic-muenchen.de

Vorstellung MIC GmbH

- Gegründet 1996
- Schwerpunktthemen
 - SAP
 - Basis, Netweaver
 - Oracle
 - Datenbankdesign
 - Performance-Tuning
 - PL/SQL-Entwicklung
 - DB-Administration
- 22 Mitarbeiter
 - SAP 15
 - Oracle 7

Allgemeines

- Für die Entwicklung und Tests großer Softwaresysteme ist es von erheblichem Vorteil, wenn in den Entwicklungs- und Testumgebungen produktionsnahe Testdaten zur Verfügung stehen.
- Nach den vielen Skandalen der letzten Jahre im Bereich Datenklau kann es sich heutzutage (eigentlich) keine Firma mehr leisten, produktive Daten in seinen Testsystemen unverändert einzuspielen oder an ihre Softwarehersteller weiterzugeben.
- Seit einigen Jahren bieten mehrere Software-Firmen Tools zur Anonymisierung von Daten an so auch Oracle mit seinem Data Masking Pack.

Grundlegende Anforderungen

- Bereitstellung von **realistischen aber nicht realen Daten**, z.B. echte Telefonnummern, gültige Kreditkartennummern, inkl. Prüfziffer und Checksumme, echte Namen statt zufälliger Zeichenketten, etc.
- **Unumkehrbarkeit**, d.h. unter keinen Umständen darf es auf der Zielumgebung möglich sein, auf die originalen Daten zurück zu schließen. Ein wesentliches Kriterium dafür ist die:
- **Vollständigkeit**, d.h. die Identifikation und Maskierung redundanter und zusammengesetzter Spalten.
- Erhalt der **Datenqualität**.
 - Maskierung **abgeleiteter Spalten**, z.B. Sozialversicherungsnummern bzw. Matchcode-Spalten
 - Erhalt der **referentiellen Integrität**, d.h. Beziehungen zwischen den Datenbanktabellen müssen auch nach der Maskierung identisch aufgelöst werden können.
- Erhalt der **Datenverteilung**, d.h. durch die Datenanonymisierung darf sich das Verhalten des Optimizers bei der Abarbeitung der SQL-Befehle nicht ändern.

Oracle Data Masking Pack

- Lizenzpflichtige Option für den Oracle Enterprise Manager
- Graphische Benutzeroberfläche
- Vordefinierte Maskierungsformate, z.B. Kreditkartennummer.
- Vordefinierte Maskierungsalgorithmen, z.B. Mischen, Löschen, Leeren, etc.
- Berücksichtigung redundanter Spalten
- Generierung ablauffähiger Anonymisierungsskripte
- Sehr gute bildschirmpezifische Online-Hilfe
- Im Projekt verwendete Version 10.204

Notwendige Voraussetzungen

- Einrichtung eines DB-Schemas mit folgenden Rechten:
 - DBA-Rolle
 - DBMS_RANDOM
 - DMBS_CRYPTO
- Unter diesem DB-Schema werden:
 - Maskierungsdefinitionen vorgenommen
 - die generierten Skripte ausgeführt

Maskierungsdefinition erstellen

Abbrechen OK

* Name

* Datenbank

Beschreibung

Spalten

Fügen Sie Spalten hinzu, die maskiert werden sollen, und definieren Sie das Maskierungsformat für jede Spalte. Fremdschlüsselspalten werden automatisch hinzugefügt, um die referenzielle Integrität aufrechtzuerhalten. Für abhängige Spalten sind keine FOREIGN KEY Constraints definiert, sie referenzieren jedoch eine maskierte Spalte aufgrund von Constraints auf Anwendungsebene. Sie können abhängige Spalten manuell zu einer maskierten Spalte hinzufügen. Wenn Sie eine Spalte aus dieser Liste entfernen, werden alle Fremdschlüssel und abhängigen Spalten entfernt.

Hinzufügen

Auswählen	Eigentümer	Tabelle	Spalte	Spaltengruppe	Datentyp	Format	Fremdschlüsselspalten	Abhängige Spalten	
								Anzahl	Hinzufügen
Keine Spalten hinzugefügt									

Fremdschlüsselspalten

Eigentümer	Tabelle	Spalte	Übergeordneter Eigentümer	Übergeordnete Tabelle	Übergeordnete Spalte
Keine Fremdschlüsselspalten					

Abhängige Spalten

Eigentümer	Tabelle	Spalte	Übergeordneter Eigentümer	Übergeordnete Tabelle	Übergeordnete Spalte	Entfernen
Keine abhängigen Spalten hinzugefügt						

[Erweiterte Optionen anzeigen](#)

Abbrechen OK

Maskierungsdefinition erstellen

* Name
 * Datenbank
 Beschreibung

Abbrechen OK

Spalten

Fügen Sie Spalten hinzu, die maskiert werden sollen, und definieren Sie das Maskierungsformat für jede Spalte. Fremdschlüsselspalten werden automatisch hinzugefügt, um die referenzielle Integrität aufrechtzuerhalten. Für abhängige Spalten sind keine FOREIGN KEY Constraints definiert, sie referenzieren jedoch eine maskierte Spalte aufgrund von Constraints auf Anwendungsebene. Sie können abhängige Spalten manuell zu einer maskierten Spalte hinzufügen. Wenn Sie eine Spalte aus dieser Liste entfernen, werden alle Fremdschlüssel und abhängigen Spalten entfernt.

Hinzufügen

Entfernen

[Alles auswählen](#) | [Nichts auswählen](#)

Auswählen	Eigentümer	Tabelle	Spalte	Spaltengruppe	Datentyp	Format	Fremdschlüsselspalten	Abhängige Spalten	
								Anzahl	Hinzufügen
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00VORNAM		VARCHAR2(20)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00NATLKZ		VARCHAR2(3)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00ANREDE		NUMBER(3)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00GEBNAM		VARCHAR2(30)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00AKADET		VARCHAR2(25)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00ADELST		VARCHAR2(25)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00NAME2		VARCHAR2(30)		0	0	

Für Spalten mit diesem Symbol ist kein Maskierungsformat definiert.

Fremdschlüsselspalten

Eigentümer	Tabelle	Spalte	Übergeordneter Eigentümer	Übergeordnete Tabelle	Übergeordnete Spalte
Keine Fremdschlüsselspalten					

Abhängige Spalten

Eigentümer	Tabelle	Spalte	Übergeordneter Eigentümer	Übergeordnete Tabelle	Übergeordnete Spalte	Entfernen
Keine abhängigen Spalten hinzugefügt						

Erweiterte Optionen anzeigen

Abbrechen OK

[Datenmaskierungsdefinitionen](#) > [Maskierungsdefinition erstellen](#) >

Spaltenmaske definieren

Eigentümer **LIFESTREAM**
Spalte **PADOVORNAM**

Tabelle **PA00PF**
Datentyp **VARCHAR2(20)**

Abbrechen OK

Standardmäßig werden alle Datensätze in der Tabelle unter Verwendung des angegebenen Formats maskiert. Sie können optional mithilfe von Bedingungen eine oder mehrere Teilmengen von Datensätzen identifizieren. Jede Teilmenge kann mit dem entsprechenden Maskierungsformat maskiert werden. Die Teilmengen werden in der Reihenfolge maskiert, in der sie angegeben werden. Eine Teilmenge wird nicht erneut maskiert, auch wenn sie eine nachfolgende Bedingung erfüllt.

Bedingung hinzufügen

Importformat		Formateintrag	Eigenschaften des Formateintrags					
			Eigenschaft	Wert	Eigenschaft	Wert	Sample	Entfernen
Alle einblenden Alle ausblenden		Array-Liste						
Auswählen Bedingung		Löschen						
Bedingungen		Festzahl						
Standardbedingung		Feste Zeichenfolge						
(Formateintrag hin		Nullwert						
		Nachverarbeitungsfunktion						
		Originaldaten beibehalten						
		Zufallsdezimzahlen						
		Zufallsziffern						
		Zufalszahlen						
		Zufalszeichenfolgen						
		Mischen						
		SQL-Ausdruck						
		Ersetzen						
		Teilzeichenfolge						
		Tabellenspalte						
		Leeren						
		Benutzerdefinierte Funktion						

Abbrechen OK

[Zielverzeichnis](#) | [Ziele](#) | [Deployments](#) | [Alerts](#) | [Konformität](#) | [Jobs](#) | [Berichte](#) | [My Oracle Support](#) | [Setup](#) | [Voreinstellungen](#) | [Hilfe](#) | [Abmelden](#)

Maskierungsdefinition erstellen

Abbrechen OK

* Name

* Datenbank

Beschreibung

Spalten

Fügen Sie Spalten hinzu, die maskiert werden sollen, und definieren Sie das Maskierungsformat für jede Spalte. Fremdschlüsselspalten werden automatisch hinzugefügt, um die referenzielle Integrität aufrechtzuerhalten. Für abhängige Spalten sind keine FOREIGN KEY Constraints definiert, sie referenzieren jedoch eine maskierte Spalte aufgrund von Constraints auf Anwendungsebene. Sie können abhängige Spalten manuell zu einer maskierten Spalte hinzufügen. Wenn Sie eine Spalte aus dieser Liste entfernen, werden alle Fremdschlüssel und abhängigen Spalten entfernt.

Hinzufügen

Entfernen

[Alles auswählen](#) | [Nichts auswählen](#)

Auswählen	Eigentümer	Tabelle	Spalte	Spaltengruppe	Datentyp	Format	Fremdschlüsselspalten	Abhängige Spalten	
								Anzahl	Hinzufügen
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00VORNAM		VARCHAR2(20)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00NATLKZ		VARCHAR2(3)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00ANREDE		NUMBER(3)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00GEBNAM		VARCHAR2(30)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00AKADET		VARCHAR2(25)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00ADELST		VARCHAR2(25)		0	0	
<input type="checkbox"/>	LIFESTREAM	PA00PF	PA00NAME2		VARCHAR2(30)		0	0	

Für Spalten mit diesem Symbol ist kein Maskierungsformat definiert.

Fremdschlüsselspalten

Eigentümer	Tabelle	Spalte	Übergeordneter Eigentümer	Übergeordnete Tabelle	Übergeordnete Spalte
Keine Fremdschlüsselspalten					

Abhängige Spalten

Eigentümer	Tabelle	Spalte	Übergeordneter Eigentümer	Übergeordnete Tabelle	Übergeordnete Spalte	Entfernen
Keine abhängigen Spalten hinzugefügt						

[Erweiterte Optionen anzeigen](#)

Abbrechen OK

Erweiterte Optionen verbergen

Datenmaskierungsoptionen

- Redo Log-Generierung während der Maskierung deaktivieren
- Statistik nach der Maskierung aktualisieren
- Während der Maskierung erstellte temporäre Tabellen löschen
- Verwaiste Fremdschlüsselzeilen ignorieren
- Wenn möglich Parallelausführung benutzen

Parallelitätsgrad Standardwert Wert

Zufallszahlenerstellung

Die Datenmaskierung kann bei der Generierung von Zufallszahlen sehr verlässliche Sicherheitstechniken verwenden. Dadurch dauert der Maskierungsvorgang allerdings länger. Soll bei der Maskierung Geschwindigkeit oder Sicherheit Vorrang haben?

- Geschwindigkeit hat Vorrang
- Sicherheit hat Vorrang

Pre-Maskierungsskript

Geben Sie ein SQL-Skript an, das vor der Maskierung ausgeführt werden soll.

Post-Maskierungs-Skript

Geben Sie ein SQL-Skript an, das nach Abschluss der Maskierung ausgeführt werden soll.

Abbrechen OK

Verarbeitung: Datenmaskierungsskript generieren

Datenbank **IE7LIFE**
Angemeldet als **voran_batch_adv**

Anzahl von Tabellen **1**
Spalten **3**

Abbrechen

Das Maskierungsskript wird generiert. Dieser Prozess kann bis zu 15 Minuten dauern.



Abbrechen

Informationen

Generierung des Datenmaskierungsskripts erfolgreich abgeschlossen.

Ergebnisse der Skriptgenerierung: MASKING_BSP

Datenbank **IE7LIFE**
Angemeldet als **voran_batch_adv**

Anzahl von Tabellen **1**
Spalten **3**

Zurück

Skriptoptionen

Verwenden Sie das Skript zum Clonen und Maskieren der Datenbank.

Clonen und maskieren

Planen Sie den Datenmaskierungs-Job. Das Skript wird vom Job ausgeführt, um den Maskierungsvorgang durchzuführen.

Job planen

Skript

Die Skriptzusammenfassung besteht aus einer Liste der Datenbankbefehle, mit denen die ausgewählten Spalten maskiert werden. Das vollständige Skript ist ein PL/SQL-Skript, das Funktionen, Prozeduren und andere Befehle umfasst, die während der Maskierung benötigt werden. Das vollständige Skript wird vom Job ausgeführt, um die Maskierung durchzuführen.

Vollständiges Skript speichern

Anzeigen Skriptzusammenfassung Vollständiges Skript

```
-- Zieldatenbank: IE7LIFE
-- Skript generiert um: 07-OKT-2011 10:00
COMMIT
ALTER SESSION ENABLE PARALLEL DML
DECLARE
CURSOR fk_sql IS select refr.owner, refr.table_name, refr.constraint_name
from dba_constraints refrd, dba_constraints refr
where refrd.owner = 'LIFESTREAM_DEVKLOM' and
refr.table_name = 'PA00PF' and
refr.constraint_type = 'R' and
refr.owner = refrd.owner and
refr.constraint_name = refrd.constraint_name;
BEGIN
FOR fk IN fk_sql
LOOP
EXECUTE IMMEDIATE 'ALTER TABLE "' || fk.owner || '"."' ||
fk.table_name || '" DROP CONSTRAINT "' || fk.constraint_name || '"';
```

Die generierten Skripte I

- Ausführung
 - direkt über den im Data Masking Pack integrierten Oracle Scheduler (Button "Job planen")
 - auf das Dateisystem speichern und über ein Batch-Steuerungssystem, z.B. UC4.

In diesem Fall ist ein manueller Eingriff notwendig, da:

```
set feedback off
set ....
set timing off
```

```
spool /pkg/i25lifed/oracle/dbs/masking1371.log
```

```
set escape \
-- Skript-Header-Abschnitt
-- =====
-- functions and procedures
```

Persistente Datenbankobjekte I

Nach Ausführung eines Skriptes bleiben folgende Datenbankobjekte persistent erhalten:

- **PROCEDURE mgmt\$mask_sendMsg:**
Die Prozedur dient zur Ausgabe beliebig langer Meldungen in die Log-Datei bei Ausführung des Anonymisierungsskriptes (Aufteilung der Meldung in Portionen zu 255 Zeichen).
- **PROCEDURE mgmt\$mask_errorExit:**
Die Prozedur dient zur Ausgabe von Fehlermeldungen in die Log-Datei bei Ausführung des Anonymisierungsskriptes.
- **PROCEDURE mgmt\$mask_errorExitOraError:**
Die Prozedur dient zur Ausgabe von Oracle-Fehlermeldungen in die Log-Datei bei Ausführung des Anonymisierungsskriptes.
- **PROCEDURE mgmt\$mask_checkDBAPrivs:**
Die Prozedur prüft, ob der DB-User, unter dem das Anonymisierungsskript gestartet wird, DBA-Rechte hat, d.h. ob ihm die Rolle "DBA" zugewiesen wurde.

Persistente Datenbankobjekte II

- PROCEDURE mgmt\$mask_setUpJobTable:
 - erzeugt, falls nicht vorhanden, im Schema des ausführenden DB-Users die Tabelle MGMT\$MASK_CHECKPOINT (essentiell für die Wiederholbarkeit der Anonymisierungsskripte).
 - versucht die ID des Anonymisierungsskriptes zusammen mit Prozessschritt 1 in die Tabelle MGMT\$MASK_CHECKPOINT einzutragen.

Ist die ID des Anonymisierungsskriptes aber bereits in der Tabelle vorhanden, so bedeutet das, dass das Anonymisierungsskript bei der letzten Ausführung mit Fehlern abgebrochen wurde. In diesem Fall liefert die Prozedur die Nummer des Prozessschrittes zurück, bei dem der Fehler aufgetreten ist und mit dem bei Wiederanlauf fortgefahren werden muss.

Jede Prozedur des Anonymisierungsskriptes prüft als erstes nach ihrem Aufruf anhand der Tabelle MGMT\$MASK_CHECKPOINT, ob sie überhaupt an der Reihe ist und trägt sich in diesem Fall in die Tabelle ein.

Persistente Datenbankobjekte III

- PROCEDURE mgmt\$mask_deleteJobTableEntry:
Die Prozedur löscht den Skripteintrag aus der Tabelle MGMT\$MASK_CHECKPOINT, wenn das Skript erfolgreich gelaufen ist.
- PROCEDURE mgmt\$mask_setStep:
Die Prozedur speichert die Nummer des aktuellen Prozessschrittes im Anonymisierungsskript in der Tabelle MGMT\$MASK_CHECKPOINT. So kann von außen verfolgt werden, welcher Prozessschritt des Anonymisierungsskriptes gerade ausgeführt wird.
- Package mgmt\$mask_util:
Package mit Hilfsprozeduren, z.B. Zufallszahlengenerierung, die für die verschiedenen Anonymisierungsstrategien gebraucht werden.
- Tabelle mgmt\$mask_checkpoint:
 - SCRIPT_ID: Nummer des Anonymisierungsskriptes (wird bei der Generierung vergeben)
 - LAST_STEP: Nummer des Prozessschrittes, bzw. der Prozedur innerhalb des Anonymisierungsskriptes, die aktuell ausgeführt wird, bzw. in der ein Fehler aufgetreten ist und abgebrochen wurde.

Grundsätzlicher Ablauf I

Grundsätzliche Schritte bei einer Tabellenanonymisierung:

1. Löschen aller Trigger, die auf dieser Tabelle definiert sind (pro Trigger 1 statische Prozedur).
2. Löschen aller Foreign-Key-Constraints, die sich auf diese Tabelle beziehen (generische Prozedur).
3. Entzug aller Zugriffsberechtigungen ("REVOKE <privilege> ON <table_name> FROM..."), die auf diese Tabelle vergeben sind (pro Privileg und Empfänger der Zugriffsberechtigung 1 statische Prozedur).
4. Löschen aller Constraints, die auf dieser Tabelle definiert sind, ohne die Foreign-Key-Constraints von dieser Tabelle auf andere (generische Prozedur).
5. Löschen aller Indizes, die auf dieser Tabelle definiert sind (pro Index 1 statische Prozedur).
6. Umbenennung der Tabelle in `<table_name> => <table_name>$DMASK`

Grundsätzlicher Ablauf II

7. Erzeugung der anonymisierten Daten aus der umbenannten Originaltabelle
("CREATE TABLE <table_name> ... AS SELECT ... FROM <table_name>\$DMASK WHERE...").
8. Restaurierung aller Default-Werte der Tabellenspalten (pro Default-Werte 1 statische Prozedur).
9. Restaurierung des Parallelisierungsgrades der Tabelle.
10. Löschen der umbenannten Originaltabelle
("DROP TABLE <table_name>\$DMASK").
11. Restaurierung aller Indizes (pro Index 1 statische Prozedur).
12. Restaurierung des Primary-Key-Constraints.
13. Restaurierung aller Unique-Key-Constraints (pro Constraint 1 statische Prozedur).
14. Restaurierung aller Check-Constraints (pro Constraint 1 statische Prozedur).

Grundsätzlicher Ablauf III

15. Restaurierung aller NOT NULL-Spalten (pro Spalte 1 statische Prozedur).
16. Restaurierung aller Foreign-Key-Constraints (pro Constraint 1 statische Prozedur).
17. Restaurierung aller Zugriffsberechtigungen ("GRANT <privilege> ON <table_name> TO..."), die auf diese Tabelle vergeben waren (pro Privileg und Empfänger der Zugriffsberechtigung 1 statische Prozedur).
18. Restaurierung aller Foreign-Key-Constraints, die sich auf den Primary-Key dieser Tabelle beziehen (pro Constraint 1 statische Prozedur).
19. Restaurierung aller Trigger (pro Trigger 1 statische Prozedur).
20. Neu-Compilierung aller Views, Packages, Package Bodies, etc., die sich auf diese Tabelle beziehen (pro Datenbankobjekt 1 statische Prozedur).
21. Restaurierung aller Spaltenkommentare (pro Kommentar 1 statische Prozedur)
22. Neuberechnung der Tabellenstatistiken
23. Clean Up der erzeugten PL/SQLProzeduren.

Wartung/Neugenerierung der Skripte I

- Da die generierten Anonymisierungsskripte zu 95% aus statischen Prozeduren bestehen, können sie nur als Snapshot der Tabellenkonstellation zum Zeitpunkt der Generierung angesehen werden.
- Schon geringste Abweichungen in der Tabellenkonstellation zum Zeitpunkt der Generierung von der zum Zeitpunkt der Ausführung machen eine Neugenerierung der Skripte unbedingt erforderlich.
- Um absolut sicher gehen zu können, müssten die Skripten eigentlich vor jeder Ausführung neu generiert werden.

Wartung/Neugenerierung der Skripte II

- Änderungen an der Tabellenstruktur
 - Neue Spalten:
keine Fehler bei Ausführung des Skriptes, aber nach Erzeugung der anonymisierten Daten aus der umbenannten Originaltabelle (Schritt 7) existieren die neuen Spalten in der anonymisierten Tabelle nicht mehr.
 - Geänderte Spalten:
 - Datentyp: Fehler im Skript bei Erzeugung der anonymisierten Daten aus der umbenannten Originaltabelle (Schritt 7).
 - Datenlänge: nicht notwendigerweise Fehler bei Ausführung des Skriptes, aber nach Erzeugung der anonymisierten Daten aus der umbenannten Originaltabelle (Schritt 7) existieren die Änderungen in der anonymisierten Tabelle nicht mehr.
 - Nachkommastellen: analog Datenlänge.
 - Nullable: keine Fehler bei Ausführung des Skriptes, aber nach Erzeugung der anonymisierten Daten aus der umbenannten Originaltabelle (Schritt 7) existieren die Änderungen in der anonymisierten Tabelle nicht mehr.
 - Defaultwerte: analog Nullable

Wartung/Neugenerierung der Skripte III

- Änderungen an der Tabellenstruktur:
 - Gelöschte Spalten:
Fehler im Skript bei Erzeugung der anonymisierten Daten aus der umbenannten Originaltabelle (Schritt 7)
- Änderungen am Index-Design:
 - Neue Indizes:
keine Fehler bei Ausführung des Skriptes aber nach Restaurierung aller Indizes (Schritt 11) sind die neue Indizes nicht mehr vorhanden.
 - Geänderte Indizes:
keine Fehler bei Ausführung des Skriptes aber nach Restaurierung aller Indizes (Schritt 11) sind die Indizes wieder so angelegt wie zum Zeitpunkt der Skriptgenerierung definiert.
 - Gelöschte Indizes:
Fehler im Skript bei Löschen aller Indizes (Schritt 5)

Wartung/Neugenerierung der Skripte IV

- Neue, geänderte oder gelöschte Grants: analog zu Indizes
- Neue, geänderte oder gelöschte Constraints: analog zu Indizes
- Neue, geänderte oder gelöschte Trigger: analog zu Indizes
- Views:
 - Neue Views:
invalid, da durch Anonymisierungsskript nicht neu compiliert.
 - Gelöschte Views:
Fehler im Skript bei Schritt 20.
- Packages: analog zu Views

Bemerkungen:

- Da im Projekt nicht vorhanden können keine Aussagen zum Ergebnis der Skriptgenerierung gemacht werden bei:
 - Partitionierten Tabellen
 - Materialized Views

Wiederholbarkeit/Übertragbarkeit

- Wiederholbarkeit der Anonymisierungsskripte
 - Bei Abbruch eines Anonymisierungsskriptes im Fehlerfall ist in der Tabelle MGMT\$MASK_CHECKPOINT die Script-ID und der Prozessschritt (Spalte LAST_STEP) gespeichert, in dem der Fehler aufgetreten ist.
 - Bei Neustart des Skriptes werden alle Prozessschritte übersprungen, deren Prozessschrittnummer STEP_NUM <> LAST_STEP ist und der Prozess mit Schritt STEP_NUM = LAST_STEP fortgeführt.
 - Tritt bei der Ausführung eines Prozessschrittes ein Fehler auf, so wird die skriptglobale Variable STEP_NUM = -1 gesetzt, so dass alle nachfolgenden Prozessschritte nicht mehr ausgeführt werden.
- Übertragbarkeit der Anonymisierungsskripte
 - Die Anonymisierungsskripte sind nur auf eine DB-Umgebungen übertragbar, die absolut identisch zu der DB-Umgebung sind, auf der die Maskierungsdefinition vorgenommen wurde.

Dont's

- Data Masking Pack => Maskierungsdefinitionen => erweiterte Optionen:
Option "Während der Maskierung erstellte temporäre Tabellen löschen":
niemals aktivieren:
 - Wenn die temporär erstellten Tabellen am Ende des Anonymisierungsskriptes gelöscht werden, ist die Wiederholbarkeit des Skriptes verloren.
 - Die temporär erstellten Tabellen werden bei der Erzeugung der anonymisierten Tabelle ("CREATE TABLE <tablename> FROM <tablename>\$DMASK ...") gejoint.
 - Tritt bei diesem Schritt ein Fehler auf, ist der CREATE-TABLE-Befehl nicht mehr ausführbar, da bei Neuanlauf die Erzeugung der temporären Tabellen übersprungen wird.
- Niemals mehr als eine Spalte pro Tabelle durch bedingtes Mischen anonymisieren:
 - Beim bedingten Mischen werden temporär erstellte Tabellen verwendet, die bei der Erzeugung der anonymisierten Daten zu einem „Merge Join Cartesian“ führen.

Bekannte Fehler

- Die Generierung der Skripte funktioniert nicht auf Datenbankschemen, die einen "." im Namen haben.
- PROCEDURE mgmt\$mask_checkDBAPrivs:
Exception-Handling funktioniert nicht, weil die Exception "NO_DATA_FOUND" nicht explizit behandelt wird.
- Die Reihenfolge bei der Neu-Compilierung in den Anonymisierungsskripten ist zufällig und berücksichtigt nicht die Abhängigkeiten der Datenbankobjekte untereinander.
Es kommt oftmals vor, dass im Anonymisierungsskript der Package Body vor der Package Specification neu compiliert wird. Dabei kann es zum Fehler "ORA-24344: success with compilation error" kommen. Das Problem ist im My Oracle Support bekannt (Bug 10243572) und der angegebene Workaround funktioniert.

Vorteile des Data Masking Packs:

- Eingängige und komfortable graphische Oberfläche:
Von der Installation bis zum 1. generierten Maskierungsskript dauerte es ca. 1 Stunde.
- Überdurchschnittlich gute Online-Hilfe
- Umfangreiche mitgelieferte Maskierungsbibliotheken
- Zahlreiche mitgelieferte Maskierungsstrategien
- Möglichkeit zur Definition virtueller Foreign-Keys
- Berücksichtigung abhängiger Spalten
- Einfache Einbindung selbstgeschriebener Maskierungsstrategien möglich
- Einfache Einbindung von Pre- und Postprocessing-Skripten

Nachteile des Data Masking Packs:

- Generierte Skripten sind viel zu statisch. Generische Prozeduren zum Wiederaufbau der Indizes, Constraints, etc. sind unumgänglich.
 - Die Notwendigkeit der Neugenerierung ist eigentlich vor jeder Ausführung des Skriptes erforderlich.
 - Eine Übertragbarkeit der Anonymisierungsskripte auf andere DB-Umgebungen scheitert schon an Unterschieden bei den Grants.
- Da absolut unformatiert sind die generierten Skripte ohne manuelle Überarbeitung sehr schwer lesbar.
- Der Anonymisierungsprozess selbst ist durch die RENAME/CREATE-Strategie m.E. umständlich und alleine schon wegen der Restaurierung aller Indizes zu aufwendig (=> Performance).
- Keine API zum Repository
- Geringe Möglichkeiten der Parametrisierbarkeit von außen, z.B. Name und Pfad der Spool-Datei.
- Keine Möglichkeit zusammengesetzte Felder als abhängige Felder zu definieren.
- Maskierungsdefinitionen eines Administrators können nicht von einem anderen OEM-Account eingesehen werden (=> Parallelentwicklung).

Fragen

Kontakt

Bernhard Eichhorn
MIC Management Consulting GmbH
Kronacher Straße 4
D-81549 München

Telefon: +49 (89) 680 711 61

Fax: +49 (89) 680 711 62

Mobil: +49 (172) 96 79 033

E-Mail bernhard.eichhorn@mic-muenchen.de

Internet: www.mic-muenchen.de