

Speicherung und Verarbeitung großer Objekte

Lisa Leuschner

Gliederung

1. Allgemein - Was sind LOBs
2. Typen von LOBs
 1. Interne
 2. Externe
3. Speicherung von LOBs
 1. Tabellenzellen
 2. Tabellenzeilen
 3. Temporär / Persistent
4. Besonderheiten und Einschränkungen
5. Zugriff auf LOBs

Was sind LOBs

- ◉ LOB - Large Object
- ◉ Datentypen die eine große Menge Daten halten sollen
- ◉ Geeignet für Daten die nicht direkt vom DBMS interpretiert werden können :
 - Semi-Strukturierte Daten
 - Logisch strukturiert
 - Bspw. XML
 - unstrukturierte Daten
 - Logisch nicht weiter unterteilbar
 - Bspw. Bilder

Interne Typen von LOBs

- ◉ BLOB - Binary Large Object
 - Daten im Binärformat
 - Bspw. : Bilder, Audio
- ◉ CLOB - Character Large Object
 - Zeichenketten mit einheitlichen Zeichensatz
 - Bspw. lange Zeichenketten/ XML Dokumente
- ◉ NCLOB - National Character Set Large Object
 - Äquivalent zu CLOB mit dem National Character Set

Externe Typen von LOBs

◎ BFILE - External Binary File

- Binärdatei außerhalb der eigentlichen Datenbank
- Nur Lesend verfügbar
- Bspw. für statische Daten, wie Bilder welche nicht manipuliert werden

Verwendung von LOBs für andere Datentypen

◎ VARRAY

- Array von LOBs

◎ XMLType

- XMLType Datentyp wird als CLOB Datentyp gespeichert
- Ermöglicht Speicherung Schemaloser XML Dokumente

◎ Oracle interMedia

- Nutzung für Datentypen ORDAudio, ORDDoc usw.

Speicherung von LOBs

- ◉ Zelleninhalt der Tabelle = LOB Instanz
- ◉ Instanz hat einen Locator(Zeiger) und einen Wert (eigentliches Objekt)
- ◉ 3 mögliche Zellenzustände
 - NULL (kein Wert/ Locator)
 - Empty (Kein Wert)
 - Populated

Speicherung von LOBs

- ◉ Unterschiede der Speicherung je nach Typ
 - Intern (BLOB, CLOB, NCLOB)
 - Zelle speichert Locator zum wert
 - Locator und Wert eindeutig
 - Extern (BFILE)
 - Zelle speichert Zeiger zur Datei
 - Zeiger eindeutig, aber es kann auf die selbe Datei mehrmals verwiesen werden

Speicherung von LOBs

◉ In-line

- Speicherung in der Tabellenzeile
- Wenn LOB Wert NULL

◉ Out-of-Line

- Speicherung außerhalb der Zeile
- Default Einstellung
- Ab Größe von 3964 b

Speicherung von LOBs

- ◉ LOB kann temporär oder persistent sein
 - Persistent
 - Ist in einer Tabelle gespeichert (ACID gilt)
 - Temporär
 - Nur im Kontext der aktuellen Anwendung verfügbar
 - Wird persistent wenn es in der Datenbank gespeichert wird

Regeln und Einschränkungen

⦿ Lobs können

- keine Primärspalten sein
- nicht in der Order by / Group by Klausel stehen
- nicht in SELECT Distinct oder SELECT Unique Queries genutzt werden

Zugriff auf LOBs

◉ Öffnen und Schließen von LOBs

■ Read Only Mode

- Locator & Wert können nicht vor Schließen der Instanz geändert werden

■ Read/ Write mode (BLOB, CLOB, NCLOB)

- Verschiebt Index Verwaltung der Lob Spalte bis zum Schließen der Instanz

Zugriff mittels SQL

- ◉ Einfache Anwendung wie bei üblichen Datentypen :
 - CLOB/NCLOB → CHAR, VARCHAR2, LONG
 - BLOB → RAW, LONG RAW
- ◉ Bspw. Erstellung :

```
CREATE TABLE media (  
    media_id NUMBER(6),  
    photo BLOB  
);
```

Zugriff mittels SQL

- ◉ Bsp. Einfügen / Abfragen :

```
INSERT INTO media (media_id, photo) VALUES(1, 0x322e..);  
SELECT photo FROM media WHERE media_ID = 1;
```

- ◉ Einschränkungen

- Maximal zugreifbare Datengröße begrenzt auf Buffer Größe (32767 Bytes)
- 4000 Byte Limit für Results in SQL

Zugriff mittels API

◉ Beispiel JDBC :

//CLOB erstellen

- Clob myClob = this.con.createClob();

//Schreibe Stream von Chars in das Clob Objekt

- Writer clobWriter = myClob.setCharacterStream(1);
- String str = this.readFile(fileName, clobWriter);

//Füge in Tabelle ein

- String sql = "INSERT INTO COFFEE_DESCRIPTIONS " + "VALUES(?,?)";
- pstmt = this.con.prepareStatement(sql);
- pstmt.setString(1, coffeeName);
- pstmt.setClob(2, myClob);
- pstmt.executeUpdate();

Quellen

- ⦿ http://docs.oracle.com/cd/B19306_01/appdev.102/b14249/toc.htm
- ⦿ <https://docs.oracle.com/javase/tutorial/jdbc/basics/gettingstarted.html>