
Technical Note

Accessing a Database LOB using Rep++

Overview

A BLOB (Binary Large Object), or its more generic form, a LOB (Large Object), is a large-sized file, such as an image, video, text, etc., stored in a database field. Because of their size, LOBs cannot be retrieved in a single block of data, but in a series of smaller chunks. Rep++ provides a way to seamlessly access a LOB stored in a database.

This article describes how to read or write LOBs in databases using Rep++.

The products and software mentioned in this document are trademarks, registered trademarks or trade names of their respective holders.

Accessing a Database LOB using Rep++

Drivers that support access to LOBs

The following Rep++ database drivers support access to LOBs.

- Oracle® Net/8
- SQL Server/dblib¹
- SQL Server 2005 and higher^{1,2}

Using Rep++ to access a LOB

Rep++ provides a way to create code that will work with all Rep++ drivers supporting LOB access. The RepPP.Connection.BuildLOBCommand command is used to build a set of SQL commands to:

- Initialize a list of LOBs
- Read from a list of LOBs
- Write to a list of LOBs

The generated SQL commands are different for each database. Here are the generated SQL commands to handle two LOB columns, MYLOB1 and MYLOB2 that belong to the table TESTLOB having a key column equal to KEY:

Generated SQL commands for SQL Server

Init:

```
UPDATE TESTLOB
SET MYLOB1=0x,
    MYLOB2=0x
WHERE KEY=:KEY
```

Read:

```
SELECT TEXTPTR(MYLOB1) AS MYLOB1,
       DATALENGTH(MYLOB1) ,
       TEXTPTR(MYLOB2) AS MYLOB2,
       DATALENGTH(MYLOB2)
FROM TESTLOB
WHERE KEY=:KEY
FOR READING IMAGE
```

Write:

```
SELECT TEXTPTR(MYLOB1) AS MYLOB1,
       DATALENGTH(MYLOB1) ,
       TEXTPTR(MYLOB2) AS MYLOB2,
       DATALENGTH(MYLOB2)
FROM TESTLOB
```

¹ Limit of 8000 bytes per chunk. The column must be of type IMAGE or TEXT.

² Not supported with ODBC (SQL Server 2005).

```
WHERE KEY=:KEY
FOR UPDATING IMAGE
```

Generated SQL commands for Oracle

Init:

```
UPDATE TESTLOB
SET MYLOB1=empty_blob(),
    MYLOB2=empty_blob()
WHERE KEY=:KEY
```

Read:

```
SELECT MYLOB1,
       MYLOB2
FROM TESTLOB
WHERE KEY=:KEY
```

Write:

```
SELECT MYLOB1,
       MYLOB2
FROM TESTLOB
WHERE KEY=:KEY
FOR UPDATE OF MYLOB1,MYLOB2
```

Table 1. Rep++ methods for accessing LOBs.

Methods	Definition
RepPP.SqlCursor.GetLOBLength	Return the actual LOB size.
RepPP.SqlCursor.ReadLOBPart	Read a chunk of data from a LOB.
RepPP.SqlCursor.TrimLOB	Trim the size of a LOB.
RepPP.SqlCursor.WriteLOBPart	Write a chunk of data to a LOB.

Creating a LOB from a file

The following C# code example demonstrates how to save two images into a table containing two LOB columns:

```
/// <summary>
/// Reads images from files and save them to a database table
/// </summary>
/// <param name="conn"> Database connection object</param>
/// <param name="imageFileName1">File full name of the first image</param>
/// <param name="imageFileName2">File full name of the second image</param>
/// <param name="strKey"> Key value identifying the row in the table</param>
private void WriteImagesToTable(RepPP.Connection conn,
                                string imageFileName1,
                                string imageFileName2,
                                string keyValue) {
    string strSqlCommand;
    RepPP.SqlCommand sql;

    // Initialize the BLOBs
    strSqlCommand = conn.BuildLOBCommand("TESTLOB",
```

```

        "MYLOB1,MYLOB2",
        "WHERE KEY=:KEY",
        RepPP.LOBAccessType.sdLOBInit);
using (sql = conn.SqlCommands.Open(strSqlCommand)) {
    sql.SetParameterValue("KEY",
        keyValue,
        RepPP.FieldType.sdFieldString);
    if (sql.Execute() != 0) {
        throw new System.ApplicationException(conn.ErrorMessage);
    }
}

// Fill the BLOBs
strSqlCommand = conn.BuildLOBCommand("TESTLOB",
    "MYLOB1,MYLOB2",
    "WHERE KEY=:KEY",
    RepPP.LOBAccessType.sdLOBUpdate);

sql = conn.SqlCommands.Open(strSqlCommand);
if (sql == null) {
    MessageBox.Show(conn.ErrorMessage);
} else {
    using (sql) {
        sql.SetParameterValue("KEY",
            keyValue,
            RepPP.FieldType.sdFieldString);
        if (sql.Execute(out RepPP.SqlCursor cur) == 0) {
            using (cur) {
                if (cur.Fetch() == 0) {
                    WriteImageToLOBColumn(cur, "MYLOB1", imageFileName1);
                    WriteImageToLOBColumn(cur, "MYLOB2", imageFileName2);
                }
            }
        } else {
            MessageBox.Show(conn.ErrorMessage);
        }
    }
}
conn.Commit();
}

/// <summary>
/// Save an image to a LOB column
/// </summary>
/// <param name="cur">Sql cursor</param>
/// <param name="lobName">Lob column name</param>
/// <param name="imageFileName">File full name of the image</param>
private void WriteImageToLOBColumn(RepPP.SqlCursor cur,
    string lobName,
    string imageFileName) {

    FileStream stream;
    int offset;
    int readSize;
    int writeSize;
    byte[] readBuffer;
    byte[] writeBuffer;

    cur.TrimLOB(lobName, 0);
    using (stream = new FileStream(imageFileName, FileMode.Open)) {
        offset = 0;
        readBuffer = new byte[8000];
        readSize = stream.Read(readBuffer, offset, 8000);
        while (readSize != 0) {
            if (readSize == 8000) {

```

```

        writeBuffer = readBuffer;
    } else {
        writeBuffer = new byte[readSize];
        Array.Copy(arrBytes,
            0,
            writeBuffer,
            0,
            readSize);
    }
    writeSize = cur.WriteLOBPart(lobName,
        offset,
        writeBuffer);

    if (writeSize <= 0) {
        throw new System.ApplicationException("Unable to write the blob
to the database");
    }
    offset += readSize;
    readSize = stream.Read(readBuffer, 0, 8000);
}
}
}
}

```

Creating a file from a LOB

The following C# code example demonstrates how to create two image files from a table containing two LOB columns.

```

/// <summary>
/// Reads images from database and save them to files
/// </summary>
/// <param name="conn"> Database connection object</param>
/// <param name="imageFileName1">File full name of the first image</param>
/// <param name="imageFileName2">File full name of the second image</param>
/// <param name="strKey"> Key value identifying the row in the
table</param>
private void WriteImagesToFiles(RepPP.Connection conn,
    string imageFileName1,
    string imageFileName2,
    string keyValue) {

    string strSqlCommand;
    RepPP.SqlCommand sql;

    strSqlCommand = conn.BuildLOBCommand("TESTLOB",
        "MYLOB1,MYLOB2",
        "WHERE KEY=:KEY",
        RepPP.LOBAccessType.sdLOBRead);
    sql = conn.SqlCommands.Open(strSqlCommand);
    if (sql == null) {
        MessageBox.Show(conn.ErrorMessage);
    } else {
        using (sql) {
            sql.SetParameterValue("KEY",
                keyValue,
                RepPP.FieldType.sdFieldString);
            if (sql.Execute(out RepPP.SqlCursor cur) == 0) {
                using (cur) {
                    if (cur.Fetch() == 0) {
                        WriteImageToFile(cur, "MYLOB1", imageFileName1);
                        WriteImageToFile(cur, "MYLOB2", imageFileName2);
                    }
                }
            }
        }
    }
}

```

```
        } else {
            MessageBox.Show(conn.ErrorMessage);
        }
    }
}

/// <summary>
/// Loads an image form the database and save it into a file
/// </summary>
/// <param name="cur">    Sql cursor</param>
/// <param name="lobName"> Name of the lob column containing the image</param>
/// <param name="imageFileName">Image file name</param>
private void WriteImageToFile(SqlCursor cur,
                               string lobName,
                               string imageFileName) {
    FileStream stream;
    int offset;
    int dataSize;
    int readSize;
    byte[] dataBuffer;

    using(stream = new FileStream(imageFileName,
                                  FileMode.Create)) {
        offset = 0;
        dataSize = cur.GetLOBLength(lobName);
        readSize = 8000;
        while (offset < dataSize) {
            if (offset + 8000 > dataSize) {
                readSize = dataSize - offset;
            }
            readSize = cur.ReadLOBPart(lobName,
                                      offset,
                                      readSize,
                                      out dataBuffer);

            if (readSize <= 0) {
                throw new System.ApplicationException("Unable to read the blob
from the database");
            }
            offset += readSize;
            stream.Write(dataBuffer, 0, readSize);
        }
    }
}
```