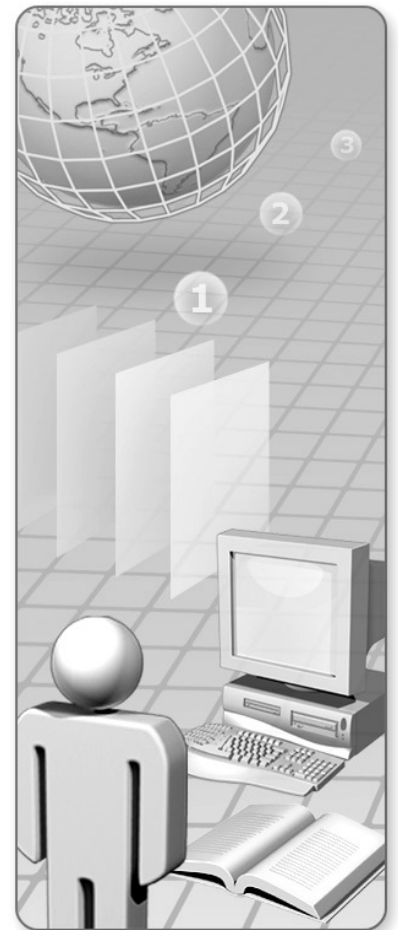


SQLHOL01: Change Data Capture

Table of Contents

Before You Begin	1
Exercise : Change Data Capture in Action	3



Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links are provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Copyright © 2007 Microsoft Corporation. All rights reserved.

Microsoft, Excel, Office, and SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Before You Begin

Estimated time to complete this lab

40 minutes

Objectives

After completing this lab, you will be able to:

- Work with Change Data Capture in SQL Server 2008

Prerequisites

Before working on this lab, you must have:

- Experience of Transact-SQL programming and SQL Server Management Studio.

Lab scenario

The objective of this Hands-on-Lab is to give you an overview of Change Data Capture.

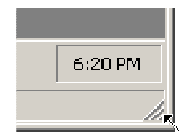
Change data capture is designed to capture insert, update, and delete activity applied to SQL Server tables, and to make the details of the changes available in an easily consumed relational format. The change tables used by change data capture contain columns that mirror the column structure of a tracked source table, along with the metadata needed to understand the changes that have occurred.

Virtual PC

This lab makes use of Microsoft Virtual PC 2007, which is an application that allows you to run multiple virtual computers on the same physical hardware. During the lab, you will use a virtual machine running Microsoft Windows Server® 2003.

Before you start the lab, familiarize yourself with the following basics of Virtual PC:

- To switch the focus for your mouse and keyboard to the virtual machine, click inside the virtual machine window.
- To remove the focus from a virtual machine, move the mouse pointer outside the virtual machine window.
- To mimic the CTRL+ALT+DELETE key combination inside a virtual machine, use <RIGHT>ALT+DEL. In Virtual PC, the <RIGHT>ALT key is called the host key.
- To enlarge the size of the virtual machine window, drag the lower-right corner of the window as seen in the screenshot.



- To switch to and from full-screen mode, press <RIGHT>ALT+ENTER.

Computers in this lab

This lab uses one computer as described in the following table. Before you begin the lab, you must start the virtual machines and then log on to the computer. In each exercise, you only have to start the virtual machine that is needed.

Virtual Machine	Computer Name	User Name	Password
SQL Server 2008 HOLs	MIAMI	Student	Pa\$\$w0rd

Start the Virtual Machine

1. Launch Microsoft Virtual PC from the Start menu or Desktop. If the Virtual PC Console does not appear, look for its icon in the System Tray, and double-click the Microsoft Virtual PC icon in the System Tray.
2. Select **SQL Server 2008 HOLs** and click **Start**.
3. When the virtual server is running, on the **Action** menu within the virtual server window, click **Ctrl+Alt+Del** (or press **Right Alt+Del** on your keyboard) to send a Ctrl+Alt+Del sequence to the login dialog box within the virtual server window.
4. Type the following information, and then click OK:
 - User name: **Student**
 - Password: **Pa\$\$w0rd**

Exercise: Change Data Capture in Action

Change data capture is designed to capture insert, update, and delete activity applied to SQL Server tables, and to make the details of the changes available in an easily consumed relational format. The change tables used by change data capture contain columns that mirror the column structure of a tracked source table, along with the metadata needed to understand the changes that have occurred..

Start SQL Server Management Studio

1. Click **Start | All Programs | Microsoft SQL Server 2008 | SQL Management Studio** to start SQL Server Management Studio.
2. Click **Connect** in the **Connect to Server** dialog box after ensuring the following settings:
 - Server type: Database Engine
 - Server name: (local)
 - Authentication: Windows Authentication
3. Click on **'File | Open | File.**
4. Browse to the C:\SQHOLS folder and open the Labscript.sql file in the Change Data Capture folder

Enable Change Data Capture

1. Review and **Highlight** the following code and click **Execute**:

```
CREATE DATABASE SQL2008CDC
GO
USE SQL2008CDC;
GO
EXEC sp_cdc_enable_db_change_data_capture;
GO
USE SQL2008CDC
GO
CREATE TABLE dbo.Employee(
EmpID int Primary Key NOT NULL,
EmpName nvarchar(100) NOT NULL,
EmpEmail nvarchar(100) NOT NULL)
GO

EXEC sp_cdc_enable_table_change_data_capture 'dbo', 'Employee', @role_name = NULL,
@supports_net_changes =1;
GO
```

2. Review and **Highlight** the following code and click **Execute**:

```
use SQL2008CDC
GO
INSERT INTO dbo.Employee
values (1, N'John Wills', N'jw@contoso.com')
GO
```

3. Review and **Highlight** the following code and click **Execute**:

```
use SQL2008CDC
GO
UPDATE dbo.Employee SET EmpName = N'Don Miles' WHERE EmpID = 1;
GO SQL2008CDC
GO
```

Review the change data

1. Review and **Highlight** the following code and click **Execute**:

```
select * from cdc.dbo_Employee_CT;
```

You will notice that only that 3 records are displayed

You will notice that there are 3 records in the resultset

Note: As mentioned earlier, in order to capture the data changes, there is a new table created in the 'cdc' schema named <schema>_<table>_CT. In this case, it contains all of the columns from the table, as well as other information including pointers to the log and the type of operation that was captured.

For the __\$operation column, 1 is a delete, 2 is an insert, 3 is the before image of an update, and 4 is the after image of an update. (SELECTs are not captured.) So, we can see our insert being captured in the first row, and then the last two rows show the before and after image of the update. Minimize SQL Server Management Studio.

2. Review and **Highlight** the following code and click **Execute**:

```
use SQL2008CDC
GO
DECLARE @from_lsn binary(10), @to_lsn binary(10);
SET @from_lsn = sys.fn_cdc_get_min_lsn('dbo_employee');
SET @to_lsn = sys.fn_cdc_get_max_lsn();
SELECT * FROM cdc.fn_cdc_get_all_changes_dbo_employee(@from_lsn, @to_lsn, 'all');
GO
```

You will notice that only that 2 records are displayed (one for insert and another for update operation)

Note: When querying for all changes in a specified interval, the row filter option 'all' is commonly used. This option returns all changes, but for an operation, only the after image is returned. To retrieve both the row containing the values before the update and the row containing the values after the update, the row filter option 'all update old' should be used.

The caller can retain the value of @to_lsn so that it can be used to determine an appropriate range for a subsequent request.

Let's understand the query:

cdc.lsn_time_mapping table is created when we enable CDC as shown in the previous exercise. It returns one row for each transaction having rows in a change table. This table is used to map between log sequence number (LSN) commit values and the time the transaction committed.

We can use the sys.fn_cdc_map_lsn_to_time and sys.fn_cdc_map_time_to_lsn system functions to query the table. One strategy is to begin by querying for the minimum and maximum LSN values for the capture instance. As shown in the following example, this method returns all changes associated with the current change data capture timeline for the capture instance.

3. Review and **Highlight** the following code and click **Execute**:

```
use SQL2008CDC
GO
DECLARE @from_lsn binary(10), @to_lsn binary(10);
SET @from_lsn = sys.fn_cdc_get_min_lsn('dbo_employee');
SET @to_lsn = sys.fn_cdc_get_max_lsn();
SELECT * FROM cdc.fn_cdc_get_net_changes_dbo_employee(@from_lsn, @to_lsn, 'all');
```

You will notice that only 1 record is returned which is the final (net) change (only the updated operation)

4. Review and **Highlight** the following code and click **Execute**:

```
use SQL2008CDC
GO
DECLARE @from_lsn binary(10), @to_lsn binary(10);
SET @from_lsn = sys.fn_cdc_get_min_lsn('dbo_employee');
SET @to_lsn = sys.fn_cdc_get_max_lsn();
SELECT * FROM cdc.fn_cdc_get_all_changes_dbo_employee(@from_lsn, @to_lsn, 'all')
SQL2008CDC
GO
```

Note: You will notice that both the insert and update operations are being shown.

5. Review and **Highlight** the following code and click **Execute**:

```
use SQL2008CDC
GO
```

```

DECLARE @end_time datetime;
DECLARE @to_lsn binary(10);
SET @end_time = GETDATE();
SELECT @to_lsn = sys.fn_cdc_map_time_to_lsn('largest less than or equal',
@end_time);
exec sys.sp_cdc_cleanup_change_table @capture_instance = 'dbo_employee',
@low_water_mark=@to_lsnSQL2008CDC
GO

```

sp_cdc_cleanup_change_table removes rows from the change table in the current database based on the specified low_water_mark value.

This stored procedure is provided for users who want to directly manage the change table cleanup process. Caution should be used, however, because the procedure affects all consumers of the data in the change table.

Cleaning CDC table and disabling CDC

6. Review and **Highlight** the following code and click **Execute**:

```

use SQL2008CDC
GO
DECLARE @from_lsn binary(10), @to_lsn binary(10);
SET @from_lsn = sys.fn_cdc_get_min_lsn('dbo_employee');
SET @to_lsn = sys.fn_cdc_get_max_lsn();
SELECT * FROM cdc.fn_cdc_get_all_changes_dbo_employee(@from_lsn, @to_lsn, 'all')

```

7. Review and **Highlight** the following code and click **Execute**:

```

use SQL2008CDC
GO
DECLARE @end_time datetime;
DECLARE @to_lsn binary(10);
SET @end_time = GETDATE();
SELECT @to_lsn = sys.fn_cdc_map_time_to_lsn('largest less than or equal',
@end_time);
exec sys.sp_cdc_cleanup_change_table @capture_instance = 'dbo_employee',
@low_water_mark=@to_lsn

```

8. Review and **Highlight** the following code and click **Execute**:

```

use SQL2008CDC
GO
DECLARE @from_lsn binary(10), @to_lsn binary(10);
SET @from_lsn = sys.fn_cdc_get_min_lsn('dbo_employee');
SET @to_lsn = sys.fn_cdc_get_max_lsn();
SELECT * FROM cdc.fn_cdc_get_all_changes_dbo_employee(@from_lsn, @to_lsn, 'all')

```

9. Review and **Highlight** the following code and click **Execute**:

```

EXECUTE sys.sp_cdc_disable_table_change_data_capture
@source_schema = N'dbo',
@source_name = N'Employee',
@capture_instance = N'dbo_Employee'

```

10. Review and **Highlight** the following code and click **Execute**:

```
USE SQL2008CDC
GO
Exec sys.sp_cdc_disable_db_change_data_capture
```

11. Close all applications and do not save changes.

12. Close Virtual PC and discard changes.

SUMMARY

To summarize, CDC facilitates an in-built mechanism to capture changes which is easy to setup and use. A good example of a data consumer targeted by this technology is an extraction, transformation, and loading (ETL) application that incrementally loads change data from SQL Server source tables to a data warehouse or data mart. Although the representation of the source tables within the data warehouse must reflect changes in those tables, an end-to-end technology that refreshes a replica of the source is not appropriate. What is needed is a reliable stream of change data, structured so that consumers can apply it to dissimilar target representations of the data. SQL Server change data capture provides this technology.