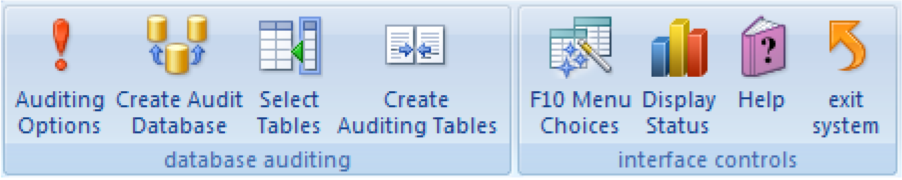
MUST: Full Database Auditing Features (FDA)

Supplementary documentation



Overview

FDA allows for full database auditing, and MUST will automatically write all the required code and modify the structure of your database to accommodate this.

Presently this feature is only available at the time a database is created, and can not be retro-fitted into existing databases.

This feature is not available in SQL server 2000. In SQL Server 2005 the feature will automatically check and set field data type mappings of Memo to (VARCHAR(MAX)) and IMAGE to (VARBINARY(MAX)) data type to support the auditing.

Standard Auditing

The standard auditing in MUST when enabled adds 4 fields to each table (this is also a prerequisite for FDA).

A_Tablename_LastUpdatedBy
A_Tablename_LastUpdatedOn
A_Tablename_CreatedBy
A_Tablename_CreatedOn

A combination of a trigger on the update event, and defaults in the fields stamp each record when created or updated with the datetime and the users windows credentials (using the `SUSER_SNAME()` function).

This is useful in giving an instant indication of who changed a record, but not what was changed or deleted. FDA addresses these issues by introducing row level auditing.

FDA

With FDA a second database is created containing a mirror image of all your desired key tables (including full support for schemas).

When any row is inserted or deleted, it is written automatically to the image table, and when a row is updated both the before image and after image of the rows are written into the image table. In addition, the LastUpdatedBy and LastUpdatedOn fields automatically change to match the user performing the operation. This ensures changes can be easily identified against a particular user.

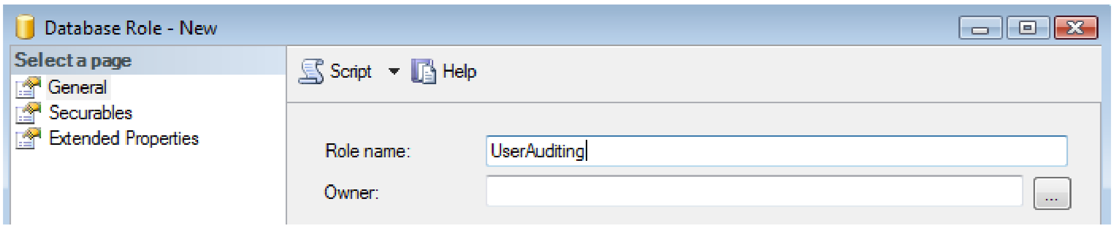
Security

All users who will be able to modify data in the main database will also require to be mapped into a security context in the auditing database, this is because if impersonating another security context to add data to the auditing database were used, the functions which identify the user would instead identify the impersonated account.

The simplest approach to this would be to create an appropriate database role in the auditing database which allows users to INSERT data, but not SELECT or any other permissions on the data. Thus users can contribute to the audit trail but now view data.

An example is show below:-

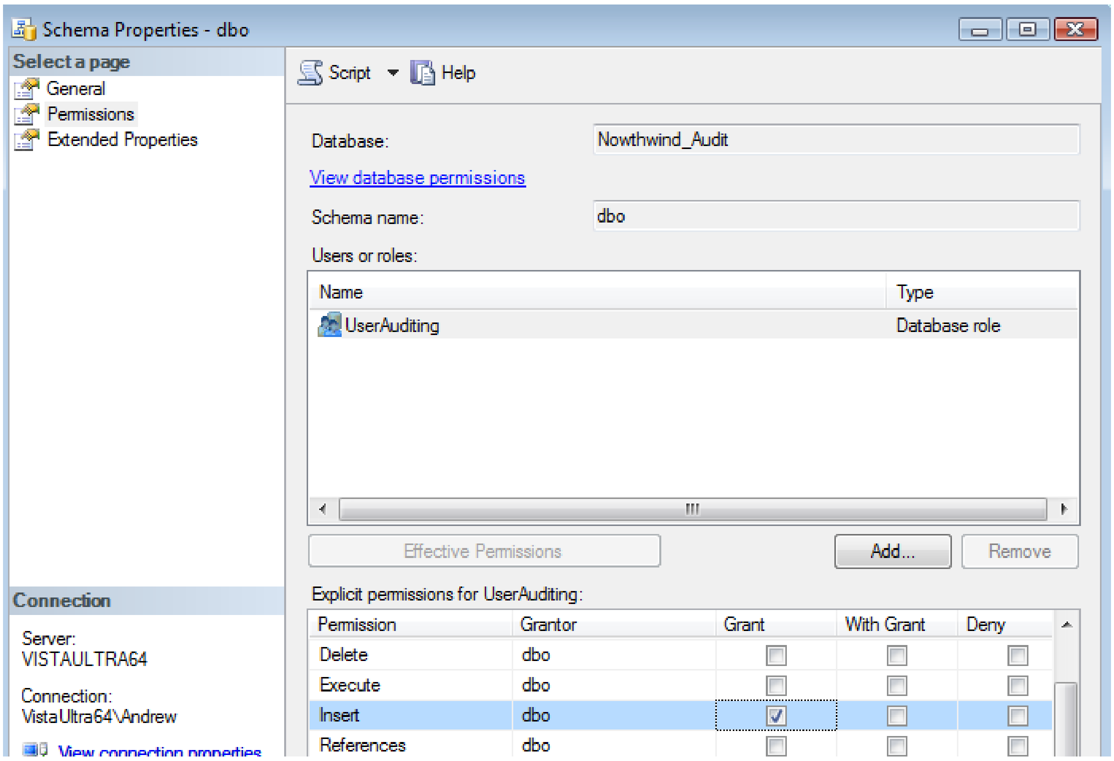
Firstly in the Auditing database create a new database role:-



Role name: UserAuditing

Owner:

The dbo schema should then be altered through the properties to grant permissions on this role as shown below.



Database: Nowthwind_Audit

Schema name: dbo

Users or roles:

Name	Type
UserAuditing	Database role

Explicit permissions for UserAuditing:

Permission	Grantor	Grant	With Grant	Deny
Delete	dbo	☐	☐	☐
Execute	dbo	☐	☐	☐
Insert	dbo	☑	☐	☐
References	dbo	☐	☐	☐

Server:
VISTAULTRA64

Connection:
VistaUltra64\Andrew

Triggers

FDA generates a BEFORE UPDATE trigger for trapping updates, an AFTER INSERT trigger for inserts and AFTER DELETE trigger for deletes.

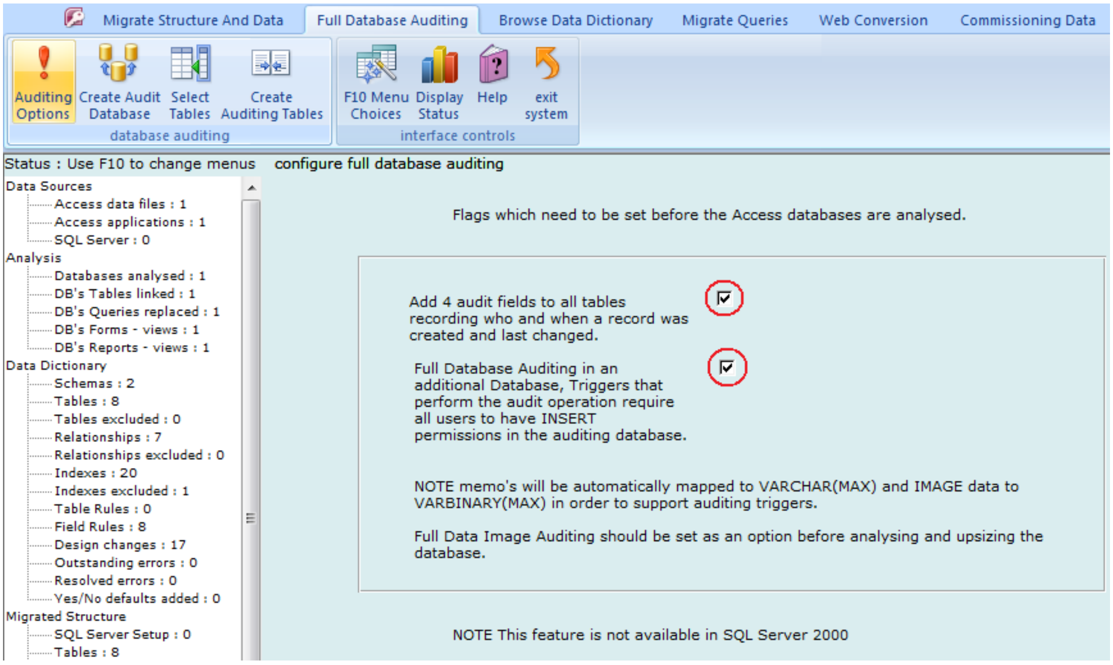
Selecting Tables

FDA can be switched on for individual tables, thus allowing a balance to be struck between audit detail and performance and storage.

Example

The following example illustrates how to configure FDA for the Northwind database.

Start with the options menu shown below; enable full database auditing before analysing your database.



Status : Use F10 to change menus

Data Sources
- Access data files : 1
- Access applications : 1
- SQL Server : 0

Analysis
- Databases analysed : 1
- DB's Tables linked : 1
- DB's Queries replaced : 1
- DB's Forms - views : 1
- DB's Reports - views : 1

Data Dictionary
- Schemas : 2
- Tables : 8
- Tables excluded : 0
- Relationships : 7
- Relationships excluded : 0
- Indexes : 20
- Indexes excluded : 1
- Table Rules : 0
- Field Rules : 8
- Design changes : 17
- Outstanding errors : 0
- Resolved errors : 0
- Yes/No defaults added : 0

Migrated Structure
- SQL Server Setup : 0
- Tables : 8

configure full database auditing

Flags which need to be set before the Access databases are analysed.

Add 4 audit fields to all tables recording who and when a record was created and last changed. ☑

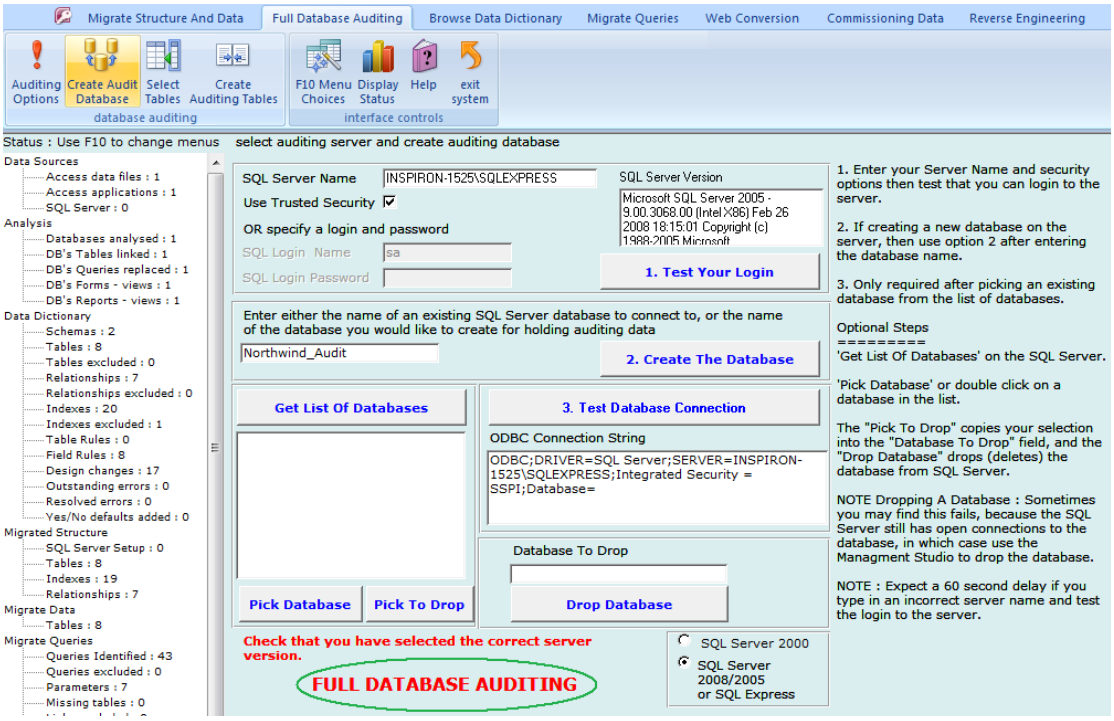
Full Database Auditing in an additional Database, Triggers that perform the audit operation require all users to have INSERT permissions in the auditing database. ☑

NOTE memo's will be automatically mapped to VARCHAR(MAX) and IMAGE data to VARBINARY(MAX) in order to support auditing triggers.

Full Data Image Auditing should be set as an option before analysing and upsizing the database.

NOTE This feature is not available in SQL Server 2000

The next step is to create the auditing database, and provide a name for the database using the “Create Audit Database” button.



Status : Use F10 to change menus

select auditing server and create auditing database

Data Sources
- Access data files : 1
- Access applications : 1
- SQL Server : 0

Analysis
- Databases analysed : 1
- DB's Tables linked : 1
- DB's Queries replaced : 1
- DB's Forms - views : 1
- DB's Reports - views : 1

Data Dictionary
- Schemas : 2
- Tables : 8
- Tables excluded : 0
- Relationships : 7
- Relationships excluded : 0
- Indexes : 20
- Indexes excluded : 1
- Table Rules : 0
- Field Rules : 8
- Design changes : 17
- Outstanding errors : 0
- Resolved errors : 0
- Yes/No defaults added : 0

Migrated Structure
- SQL Server Setup : 0
- Tables : 8
- Indexes : 19
- Relationships : 7

Migrate Data
- Tables : 8

Migrate Queries
- Queries Identified : 43
- Queries excluded : 0
- Parameters : 7
- Missing tables : 0

SQL Server Name: INSPIRON-1525\SQLEXPRESS

Use Trusted Security ☑

OR specify a login and password

SQL Login Name: sa

SQL Login Password:

SQL Server Version: Microsoft SQL Server 2005 - 9.00.3068.00 (Intel X86) Feb 26 2008 18:15:01 Copyright (c) 1988-2005 Microsoft

Enter either the name of an existing SQL Server database to connect to, or the name of the database you would like to create for holding auditing data

Northwind_Audit

ODBC Connection String: ODBC;DRIVER=SQL Server;SERVER=INSPIRON-1525\SQLEXPRESS;Integrated Security = SSPI;Database=

Database To Drop:

Check that you have selected the correct server version.

○ SQL Server 2000
● SQL Server 2008/2005 or SQL Express

1. Enter your Server Name and security options then test that you can login to the server.

2. If creating a new database on the server, then use option 2 after entering the database name.

3. Only required after picking an existing database from the list of databases.

Optional Steps
=========
'Get List Of Databases' on the SQL Server.

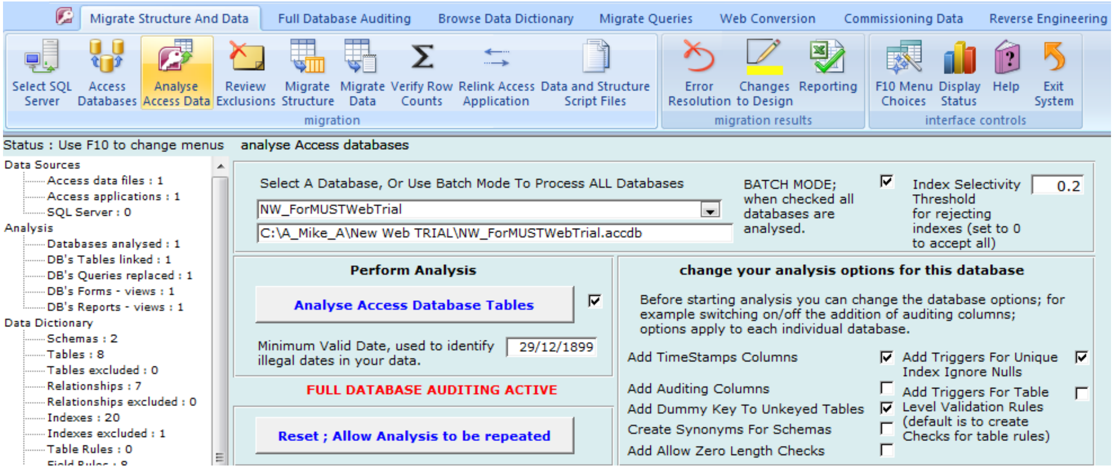
'Pick Database' or double click on a database in the list.

The "Pick To Drop" copies your selection into the "Database To Drop" field, and the "Drop Database" drops (deletes) the database from SQL Server.

NOTE Dropping A Database : Sometimes you may find this fails, because the SQL Server still has open connections to the database, in which case use the Managment Studio to drop the database.

NOTE : Expect a 60 second delay if you type in an incorrect server name and test the login to the server.

Now return to the main database menu and proceed as normal to select the database server details for upsizing, select your Access database and then analyse your database.



Status : Use F10 to change menus

analyse Access databases

Data Sources
- Access data files : 1
- Access applications : 1
- SQL Server : 0

Analysis
- Databases analysed : 1
- DB's Tables linked : 1
- DB's Queries replaced : 1
- DB's Forms - views : 1
- DB's Reports - views : 1

Data Dictionary
- Schemas : 2
- Tables : 8
- Tables excluded : 0
- Relationships : 7
- Relationships excluded : 0
- Indexes : 20
- Indexes excluded : 1
- Table Rules : 0

Select A Database, Or Use Batch Mode To Process ALL Databases

NW_ForMUSTWebTrial

C:\A_Mike_A\New Web TRIAL\NW_ForMUSTWebTrial.accdb

BATCH MODE; when checked all databases are analysed. ☑

Index Selectivity Threshold for rejecting indexes (set to 0 to accept all): 0.2

Analyse Access Database Tables ☑

Minimum Valid Date, used to identify illegal dates in your data. 29/12/1899

change your analysis options for this database

Before starting analysis you can change the database options; for example switching on/off the addition of auditing columns; options apply to each individual database.

Add TimeStamps Columns ☑

Add Auditing Columns ☐

Add Dummy Key To Unkeyed Tables ☑

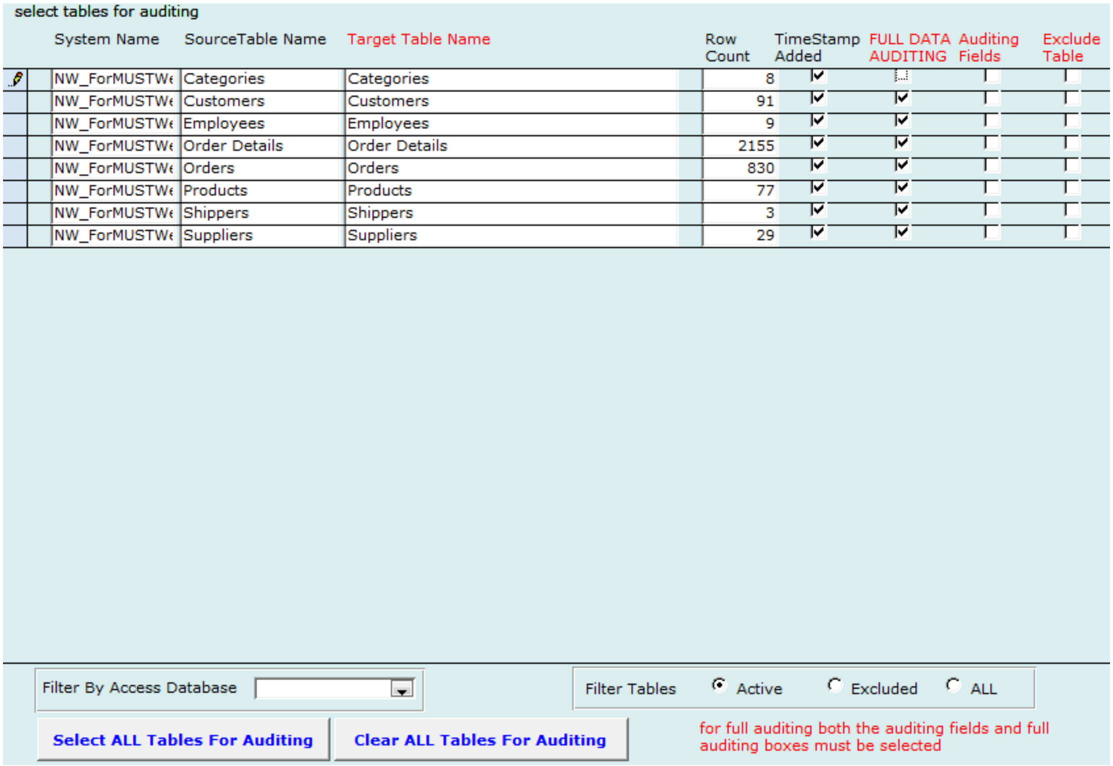
Create Synonyms For Schemas ☐

Add Allow Zero Length Checks ☐

Add Triggers For Unique Index Ignore Nulls ☑

Add Triggers For Table Level Validation Rules (default is to create Checks for table rules) ☐

Once completed, return to the options menu and use the select tables to audit button to select the target tables.



select tables for auditing

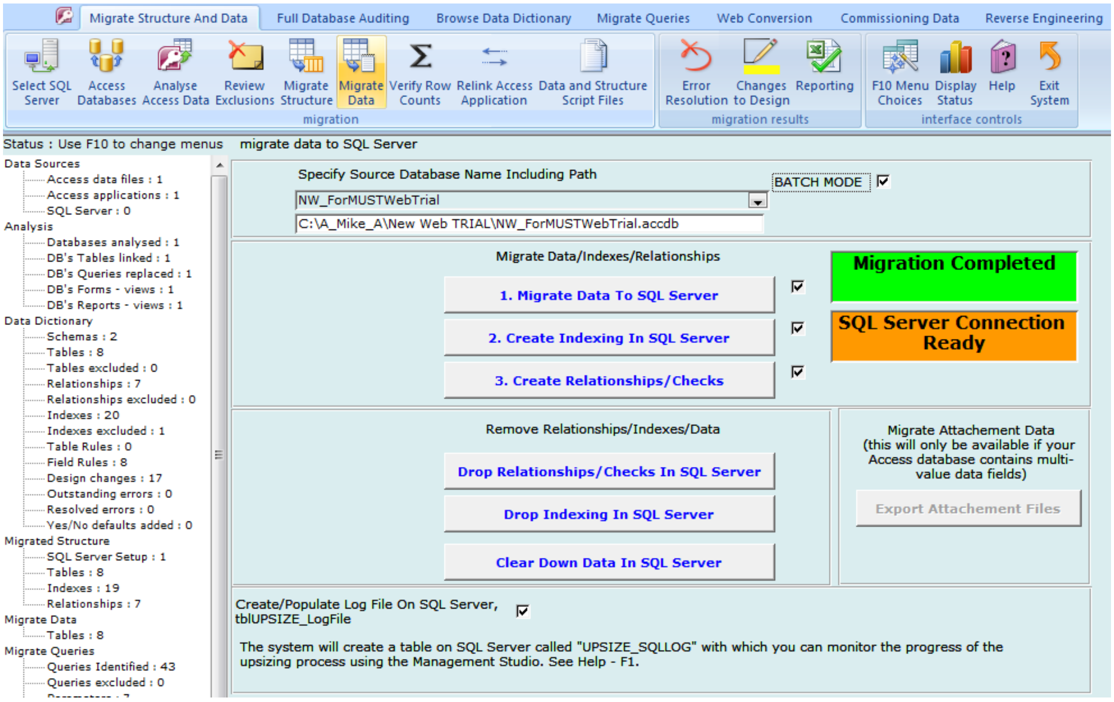
	System Name	SourceTable Name	Target Table Name	Row Count	TimeStamp Added	FULL DATA AUDITING	Auditing Fields	Exclude Table
✎	NW_ForMUSTW€	Categories	Categories	8	☑	☐	☐	☐
	NW_ForMUSTW€	Customers	Customers	91	☑	☑	☐	☐
	NW_ForMUSTW€	Employees	Employees	9	☑	☑	☐	☐
	NW_ForMUSTW€	Order Details	Order Details	2155	☑	☑	☐	☐
	NW_ForMUSTW€	Orders	Orders	830	☑	☑	☐	☐
	NW_ForMUSTW€	Products	Products	77	☑	☑	☐	☐
	NW_ForMUSTW€	Shippers	Shippers	3	☑	☑	☐	☐
	NW_ForMUSTW€	Suppliers	Suppliers	29	☑	☑	☐	☐

Filter By Access Database

Filter Tables ● Active ○ Excluded ○ ALL

for full auditing both the auditing fields and full auditing boxes must be selected

Now upsize your database, it is only when you add the relationships that the auditing triggers will be generated.



Status : Use F10 to change menus

Data Sources
- Access data files : 1
- Access applications : 1
- SQL Server : 0

Analysis
- Databases analysed : 1
- DB's Tables linked : 1
- DB's Queries replaced : 1
- DB's Forms - views : 1
- DB's Reports - views : 1

Data Dictionary
- Schemas : 2
- Tables : 8
- Tables excluded : 0
- Relationships : 7
- Relationships excluded : 0
- Indexes : 20
- Indexes excluded : 1
- Table Rules : 0
- Field Rules : 8
- Design changes : 17
- Outstanding errors : 0
- Resolved errors : 0
- Yes/No defaults added : 0

Migrated Structure
- SQL Server Setup : 1
- Tables : 8
- Indexes : 19
- Relationships : 7

Migrate Data
- Tables : 8

Migrate Queries
- Queries Identified : 43
- Queries excluded : 0

migrate data to SQL Server

Specify Source Database Name Including Path

BATCH MODE ☑

NW_ForMUSTWebTrial

C:\A_Mike_A\New Web TRIAL\NW_ForMUSTWebTrial.accdb

Migrate Data/Indexes/Relationships

1. Migrate Data To SQL Server ☑

2. Create Indexing In SQL Server ☑

3. Create Relationships/Checks ☑

Migration Completed

SQL Server Connection Ready

Remove Relationships/Indexes/Data

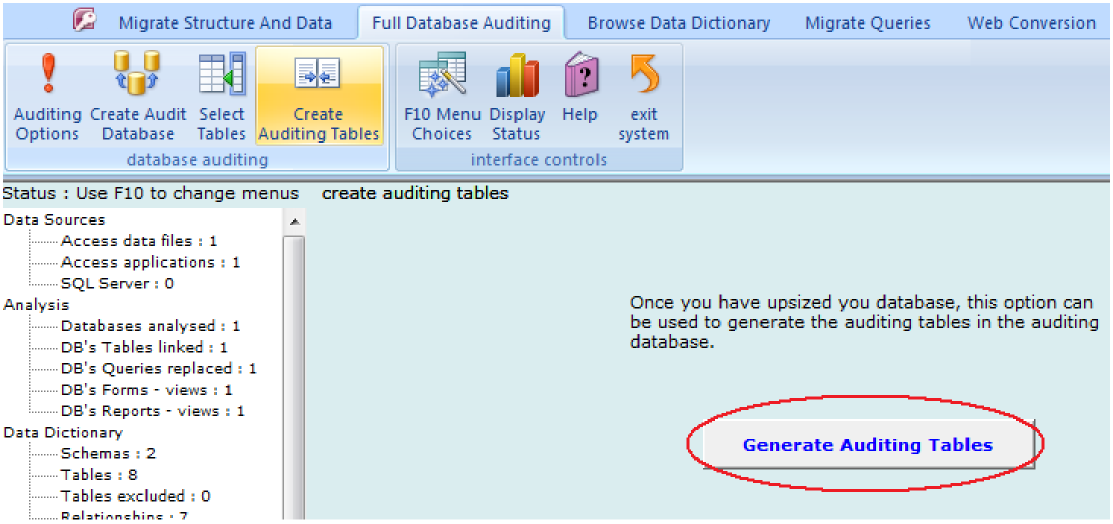
Drop Relationships/Checks In SQL Server
Drop Indexing In SQL Server
Clear Down Data In SQL Server

Migrate Attachement Data
(this will only be available if your Access database contains multi-value data fields)

Create/Populate Log File On SQL Server, tblUPSIZE_LogFile ☑

The system will create a table on SQL Server called "UPSIZE_SQLLOG" with which you can monitor the progress of the upsizing process using the Management Studio. See Help - F1.

Then return to the ‘Create Auditing Tables’ area and generate the auditing database.



Status : Use F10 to change menus create auditing tables

Data Sources
- Access data files : 1
- Access applications : 1
- SQL Server : 0

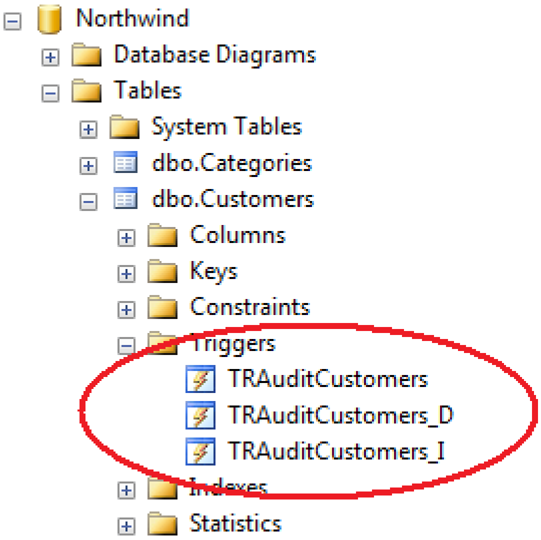
Analysis
- Databases analysed : 1
- DB's Tables linked : 1
- DB's Queries replaced : 1
- DB's Forms - views : 1
- DB's Reports - views : 1

Data Dictionary
- Schemas : 2
- Tables : 8
- Tables excluded : 0
- Relationships : 7

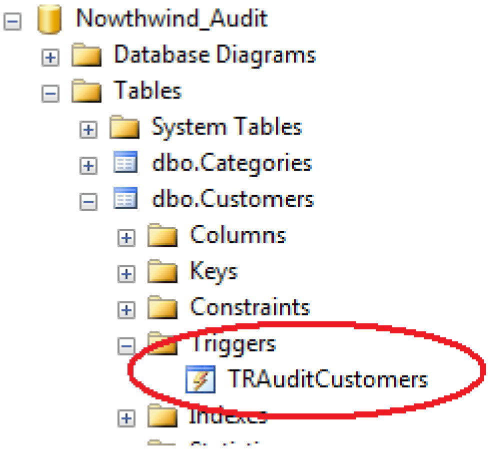
Once you have upsized you database, this option can be used to generate the auditing tables in the auditing database.

There is also an option to generate a script file for creating the auditing tables.

Each table in your main database will have three auditing triggers.



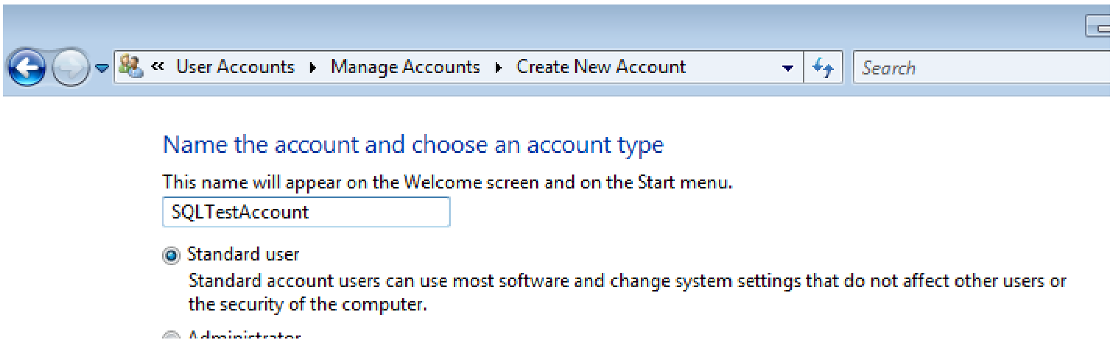
And each table in the auditing database has a single trigger.



Verifying Security

Security as ever is very important, and the following steps are aimed at assisting a developer in going through this process. These screen shorts are from Vista.

Using the control panel we create another windows account.



Name the account and choose an account type

This name will appear on the Welcome screen and on the Start menu.

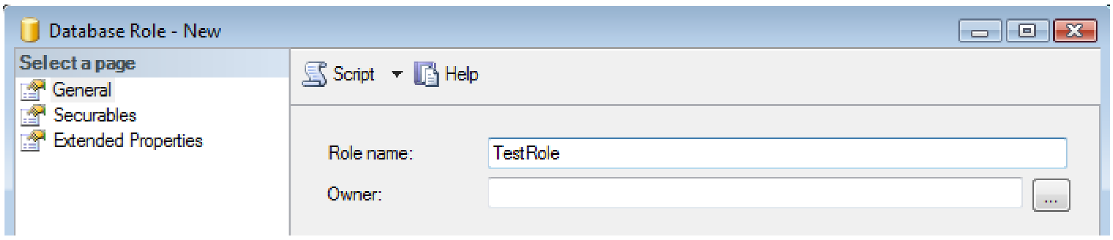
SQLTestAccount

● Standard user

Standard account users can use most software and change system settings that do not affect other users or the security of the computer.

○ Administrator

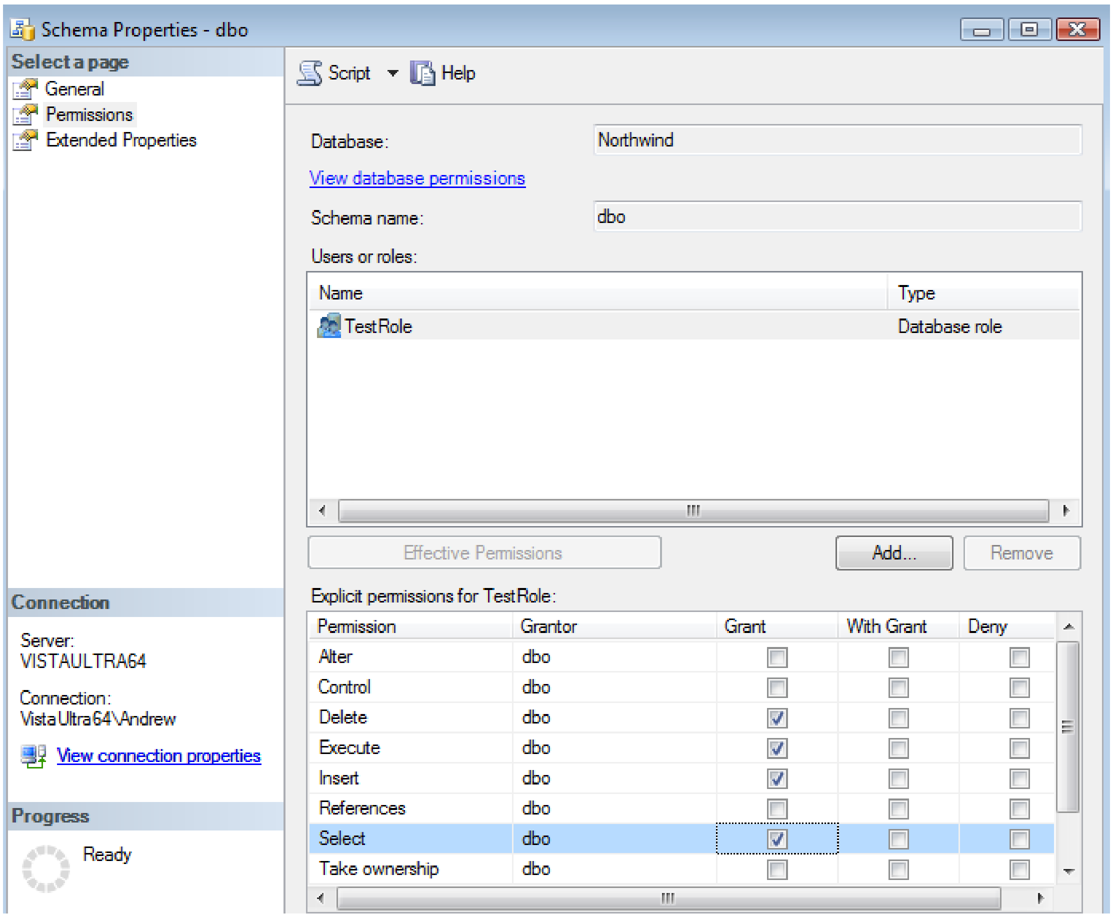
In your database create a database role.



Role name: TestRole

Owner:

Then on the dbo schema, grant Select, Insert, Update and Delete to the TestRole.



Database: Northwind

Schema name: dbo

Users or roles:

Name	Type
TestRole	Database role

Server:
VISTAULTRA64

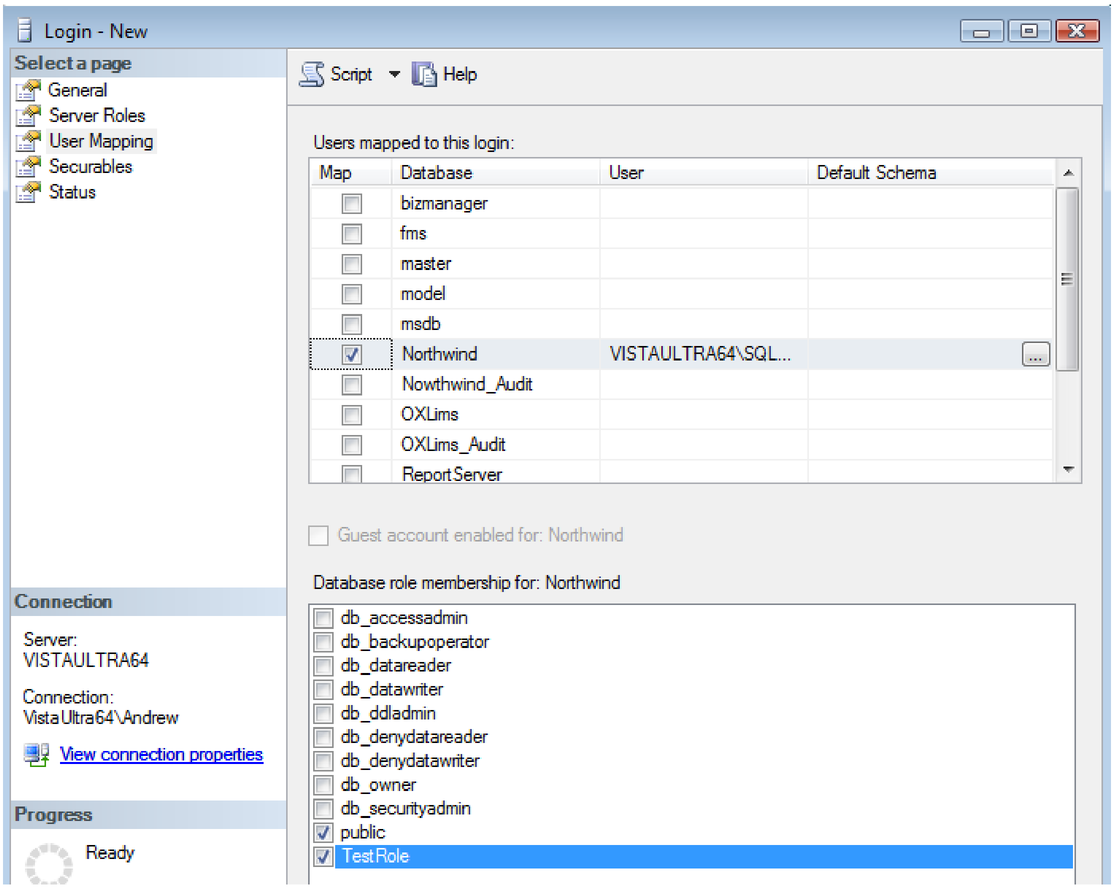
Connection:
VistaUltra64\Andrew

Explicit permissions for TestRole:

Permission	Grantor	Grant	With Grant	Deny
Alter	dbo	☐	☐	☐
Control	dbo	☐	☐	☐
Delete	dbo	☑	☐	☐
Execute	dbo	☑	☐	☐
Insert	dbo	☑	☐	☐
References	dbo	☐	☐	☐
Select	dbo	☑	☐	☐
Take ownership	dbo	☐	☐	☐

This grants this database role the ability to update data.

Next we create the SQL Server Login and assign the user to our test role.



Server:
VISTAULTRA64

Connection:
VistaUltra64\Andrew

Users mapped to this login:

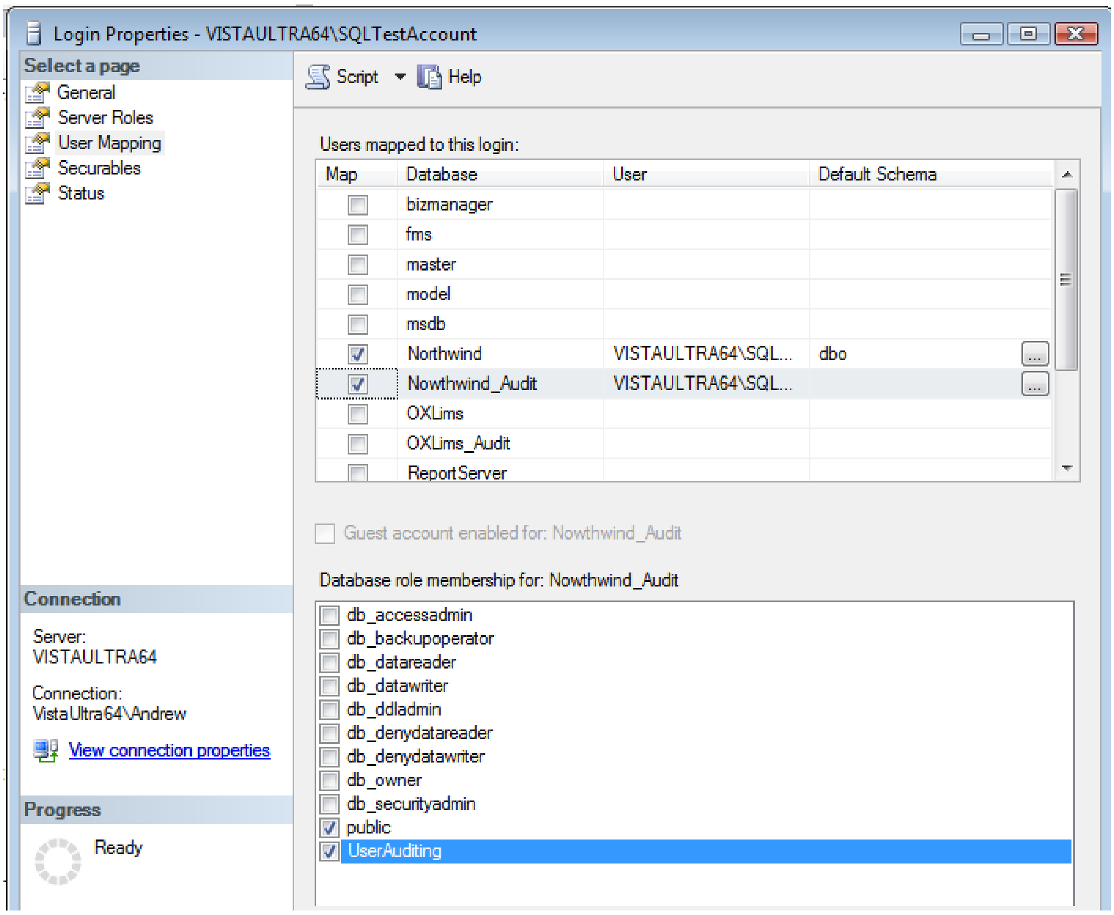
Map	Database	User	Default Schema
☐	bizmanager		
☐	fms		
☐	master		
☐	model		
☐	msdb		
☑	Northwind	VISTAULTRA64\SQL...	
☐	Nowthwind_Audit		
☐	OXLims		
☐	OXLims_Audit		
☐	ReportServer		

☐ Guest account enabled for: Northwind

Database role membership for: Northwind

- ☐ db_accessadmin
- ☐ db_backupoperator
- ☐ db_datareader
- ☐ db_datawriter
- ☐ db_ddladmin
- ☐ db_denydatareader
- ☐ db_denydatawriter
- ☐ db_owner
- ☐ db_securityadmin
- ☑ public
- ☑ TestRole

This user now also requires permissions in the Auditing database using the UserAuditing role which allows them only to contribute to data in the Audit database.



Login Properties - VISTAULTRA64\SQLTestAccount

Users mapped to this login:

Map	Database	User	Default Schema
☐	bizmanager		
☐	fms		
☐	master		
☐	model		
☐	msdb		
☑	Northwind	VISTAULTRA64\SQL...	dbo
☑	Nowthwind_Audit	VISTAULTRA64\SQL...	
☐	OXLims		
☐	OXLims_Audit		
☐	ReportServer		

☐ Guest account enabled for: Nowthwind_Audit

Database role membership for: Nowthwind_Audit

☐ db_accessadmin
☐ db_backupoperator
☐ db_datareader
☐ db_datawriter
☐ db_ddladmin
☐ db_denydatareader
☐ db_denydatawriter
☐ db_owner
☐ db_securityadmin
☑ public
☑ UserAuditing

Connection

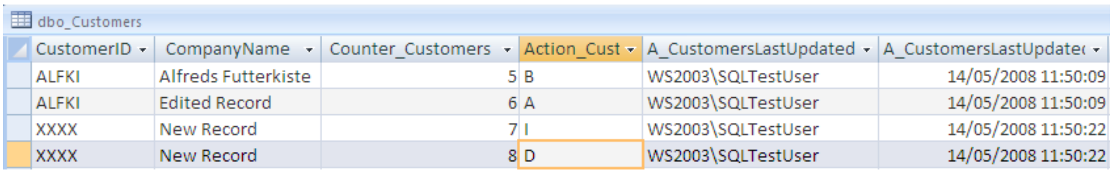
Server:
VISTAULTRA64

Connection:
VistaUltra64\Andrew

Progress

Ready

Audit Table Data



dbo_Customers

CustomerID	CompanyName	Counter_Customers	Action_Cust	A_CustomersLastUpdated	A_CustomersLastUpdate
ALFKI	Alfreds Futterkiste	5	B	WS2003\SQLTestUser	14/05/2008 11:50:09
ALFKI	Edited Record	6	A	WS2003\SQLTestUser	14/05/2008 11:50:09
XXXX	New Record	7	I	WS2003\SQLTestUser	14/05/2008 11:50:22
XXXX	New Record	8	D	WS2003\SQLTestUser	14/05/2008 11:50:22

Show above is sample data, each row has an Action_TableName field to indicate what the action was and details of who changed the record and when it was changed.

When a user inserted a record it is flagged I.
When the user edited the data we have a B (before image) and A (After Image).
When the user deletes the data we have a D record.

Note that the A_TablenameLastUpdatedBy always gets re-set to the current user who made the changes, even on the before image of the data; this allows the audited records to be easily located.

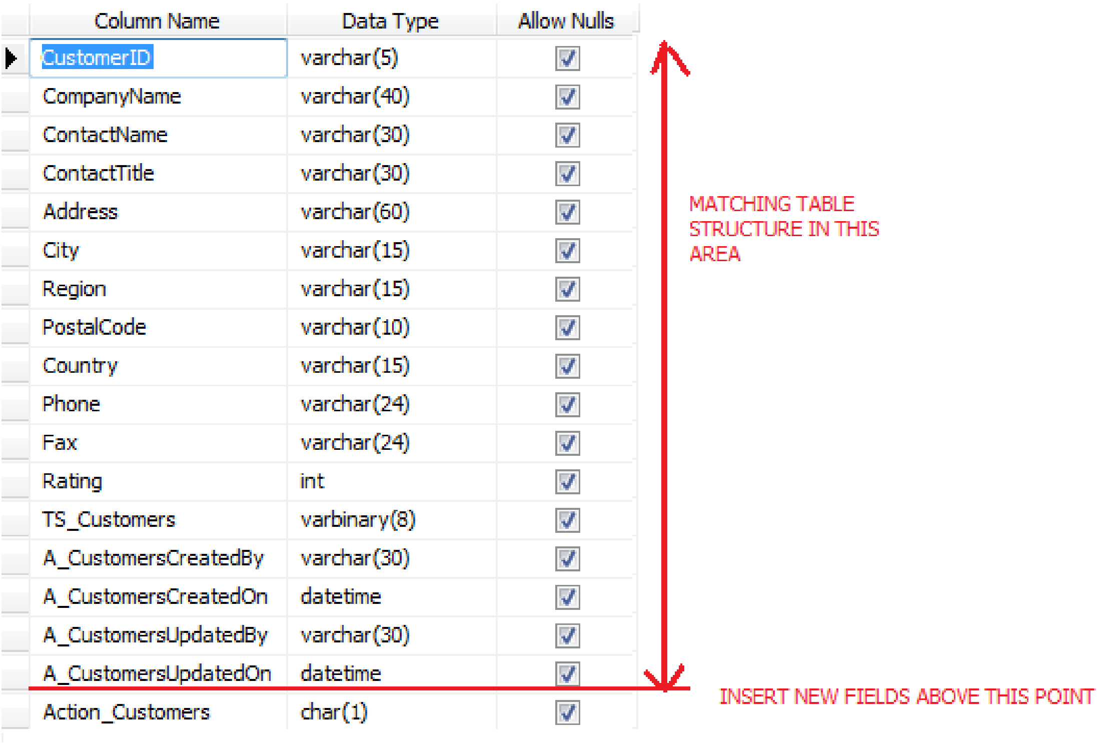
When investigating and testing the auditing a Windows 2003 or similar server is required, it appears that on Vista the authentication when switching between accounts does not behave always as anticipated.

Design Changes to Table Structure

Once auditing tables have been constructed, care must be taken when adding new fields to the databases, such that new fields are first added into the audit database and then into the live database.

When changing the fields the field order between tables in both databases must be preserved.

In particular, new fields MUST BE ADDED BEFORE THE LAST FIELDS AT THE BOTTOM OF EACH TABLE.



Column Name	Data Type	Allow Nulls
CustomerID	varchar(5)	☑
CompanyName	varchar(40)	☑
ContactName	varchar(30)	☑
ContactTitle	varchar(30)	☑
Address	varchar(60)	☑
City	varchar(15)	☑
Region	varchar(15)	☑
PostalCode	varchar(10)	☑
Country	varchar(15)	☑
Phone	varchar(24)	☑
Fax	varchar(24)	☑
Rating	int	☑
TS_Customers	varbinary(8)	☑
A_CustomersCreatedBy	varchar(30)	☑
A_CustomersCreatedOn	datetime	☑
A_CustomersUpdatedBy	varchar(30)	☑
A_CustomersUpdatedOn	datetime	☑
Action_Customers	char(1)	☑

MATCHING TABLE STRUCTURE IN THIS AREA

INSERT NEW FIELDS ABOVE THIS POINT

You will notice that the auditing tables do not have a primary key. If required you could add a key which incorporates the datetime fields.

Do NOT add a column with an IDENTITY property as a primary key, because due to a restriction in Access if the real table has an IDENTITY column, then the value returned to Access on linked tables, will be the value from the auditing table. Unfortunately Access appears to use @@IDENTITY and not SCOPE_IDENTITY() to return the key value.

Triggers, Behind the Scenes

In the auditing database the following is an example of the trigger which is added to each auditing table, to identify who made the changes to the data.

```
CREATE TRIGGER [dbo].[TR_Customers_U]
ON [dbo].[Customers]
FOR INSERT AS
UPDATE [Customers]
 SET [A_CustomersUpdatedBy] = SUSER_SNAME(), [A_CustomersUpdatedOn] = GetDate()
FROM  [Customers] s
INNER JOIN [INSERTED] i
ON s.[CustomerID] = i.[CustomerID]
AND s.[TS_Customers] = i.[TS_Customers]
```

In the main database each table contains three triggers. The first logs updates.

```
CREATE TRIGGER [dbo].[TRAuditCustomers] ON [dbo].[Customers]
FOR UPDATE
AS
UPDATE [Customers]
SET [A_CustomersLastUpdatedBy] = SUSER_SNAME(),
[A_CustomersLastUpdatedOn] = GetDate()
FROM  [Customers] s INNER JOIN [INSERTED] i
ON s.[CustomerID] = i.[CustomerID]

INSERT INTO Nowthwind_Audit.dbo.[Customers]
SELECT *,'B' FROM DELETED
INSERT INTO Nowthwind_Audit.dbo.[Customers]
SELECT *,'A' FROM INSERTED

CREATE TRIGGER [dbo].[TRAuditCustomers_D] ON [dbo].[Customers]
AFTER DELETE
AS
INSERT INTO Nowthwind_Audit.dbo.[Customers] SELECT *,'D' FROM DELETED

CREATE TRIGGER [dbo].[TRAuditCustomers_I] ON [dbo].[Customers]
AFTER INSERT
AS
INSERT INTO Nowthwind_Audit.dbo.[Customers] SELECT *,'I' FROM INSERTED
```