



ENOVIA SmarTeam

Microsoft® SQL Server 2005 Administration Guide

© Dassault Systèmes, 2004, 2008. All rights reserved.

CATIA, ENOVIA, SMARTEAM and the 3DS logo are registered trademarks of Dassault Systèmes or its subsidiaries in the US and/or other countries.

PROPRIETARY RIGHTS NOTICE: This documentation is the property of Dassault Systèmes. This documentation shall be treated as confidential information and may only be used by employees or contractors of the Customer in accordance with the terms of the End-User License Agreement accepted by Customer.

Any use of the Licensed Program contained in this media or accompanying it, is subject to the terms of the End User License Agreement accepted by Customer. The Licensed Program is protected by international copyright laws and international treaties. Unauthorized use, reproduction and/or distribution of any of the Licensed Program, or any part thereof, may result in severe civil and/or criminal penalties, and will be prosecuted to the maximum extent possible under the law. Company names and product names mentioned herein are the property of their respective owners and certain portions of the Licensed Program contain elements subject to copyright owned by these entities. See the Documentation CD provided with the Licensed Program for details and/or additional terms and conditions relating to these entities.

Part Number: EDR-A8-180007

Table of Contents

CHAPTER 1: INTRODUCTION	2
CHAPTER 2: DATABASE CONCEPTS.....	3
MSDE and SmarTeam Databases	3
Attaching and Detaching a Database	4
Connecting a SmarTeam Application to an MSDE Instance	5
Creating a Connection String	5
CHAPTER 3: ADMINISTRATIVE ACTIVITIES	6
Logging into the OSQL Utility	6
Displaying OSQL Help.....	6
Logging into OSQL	6
Creating a New MSDE SmarTeam Database	7
Creating an Empty MSDE database:.....	7
Creating a SmarTeam Database in MSDE	8
Backing Up and Restoring an MSDE Database	8
Introduction.....	8
Preparing a Backup Device	8
Backing up the Database	9
Restoring the Database from Backup	10
Restoring an MSDE Database with a Different Name.....	10
Transporting a MSDE Database	10
Detaching an MSDE Database File	11
Attaching an MSDE Database File.....	11
Renaming an Attached MSDE Database.....	12
Resolving Names Duplication when Attaching MSDE Databases	12
Security	13
User Management.....	13
Changing an MSDE Database User Password.....	13
CHAPTER 4: TROUBLESHOOTING	15
Issue: Attach fails.....	15
Issue: SmarTeam Connect fails	15
APPENDIX A: DATABASE CONNECTION.....	17

Chapter 1: Introduction

This document provides various administration procedures for Microsoft® SQL Server 2005 Express Edition in a SmarTeam implementation. It enables you to successfully setup, customize and maintain a Microsoft® SQL Server database system in a corporate environment. In this document Microsoft® SQL Server 2005 Express Edition will be referred to as MS SQL Server 2005.

Note: Any mention of MSDE in this document is also applicable to Microsoft® SQL Server 2005 Express Edition.

Related Documentation

SmarTeam Documentation

See the relevant SmarTeam Documentation CD and the SmarTeam internet site.

MSDE Documentation

See the relevant Microsoft MSDE documentation.

Chapter 2: Database Concepts

This chapter presents some basic database concepts required for understanding the operations in Chapter 3.

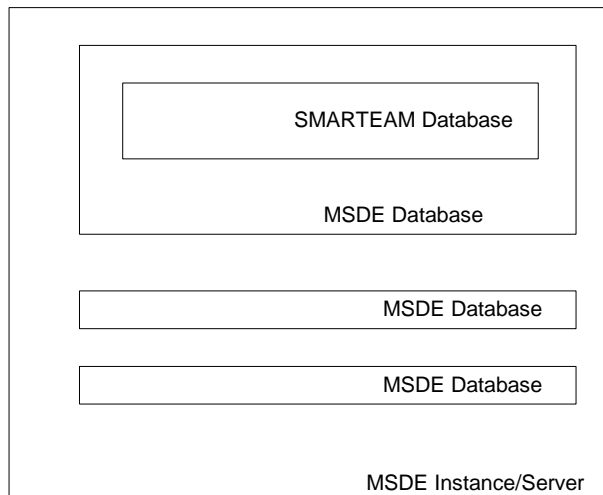
The following topics are included:

- MSDE and SmarTeam Databases
- Attaching and Detaching a Database to an MSDE Server
- Connecting a SmarTeam Application to an MSDE Instance

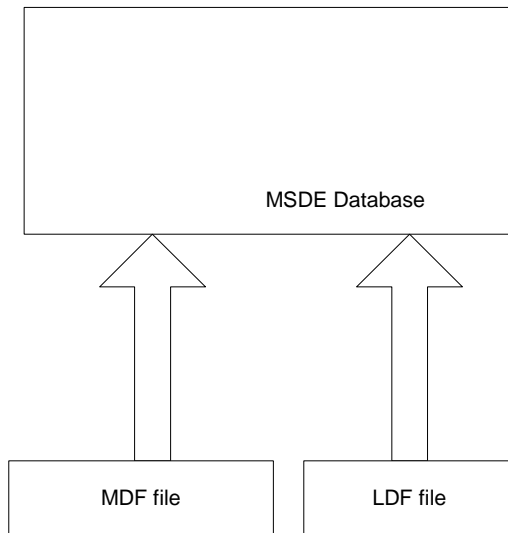
MSDE and SmarTeam Databases

The following figures show the relationship between a

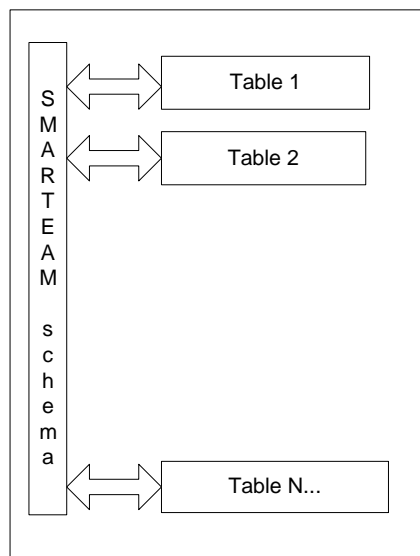
- MSDE Instance
- MSDE database attached to the MSDE Instance
- SmarTeam database contained in the MSDE database.



SMARTEAM Database inside MSDE Database
attached to MSDE Instance



MSDE Database Physical Structure



SMARTEAM Database Logical Structure

Attaching and Detaching a Database

In an MSDE Server, the data and transaction log files of a database can be detached and then reattached to another server (instance), or even to the same server. Detaching a database removes the database from MSDE Server but leaves the database intact within the data and transaction log files that compose the database. These data and transaction log files can then be used to attach the database to any instance of MSDE Server, including the server from which the database was detached. This makes the database available in exactly the same state it was in when it was detached.

Detaching and attaching databases is useful if you want to move a database:

- From one computer to another without having to re-create the database and then restore the database backup manually.

- To a different physical disk, for example, when the disk containing the database file has run out of disk space and you want to expand the existing file rather than add a new file to the database on the other disk.

To move a database, or database file, to another server or disk:

1. Detach the database.
2. Move the database file(s) to the other server or disk.
3. Attach the database specifying the new location of the moved file(s).

Connecting a SmarTeam Application to an MSDE Instance

An MSDE instance is an application, responsible for managing all its databases; in particular, it manages all communications with the databases.

A SmarTeam application must connect to an instance of MSDE Server before it can work with an MSDE Server SmarTeam database. A SmarTeam application can open multiple connections to an instance of MSDE Server.

Creating a Connection String

A SmarTeam application uses OLE DB Provider to connect to an MSDE server. The OLE DB Provider technology requires a specially configured connection string, which includes all parameters required to connect to the MSDE server. You use the SmarTeam Database Connection Manager utility to create and manage the required connection string to connect to SmarTeam databases (see Appendix).

Chapter 3: Administrative Activities

This chapter shows how to work with the command-line OSQL utility to perform basic sql administrative operations on MSDE servers. OSQL allows you to enter Transact-SQL statements, system procedures, and script files for maintaining an MSDE database.

This chapter is intended for sales, implementation or QA personnel or regular users who do not have GUI-based administrative tools for MSDE. While GUI-based administrative tools are not provided with the standard SmarTeam application, Microsoft provides the OSQL application with MSDE.

Note that there are several shareware GUI-based third-party utilities available for working with MSDE that support the same functionality as the Microsoft SQL Server administrative client tools.

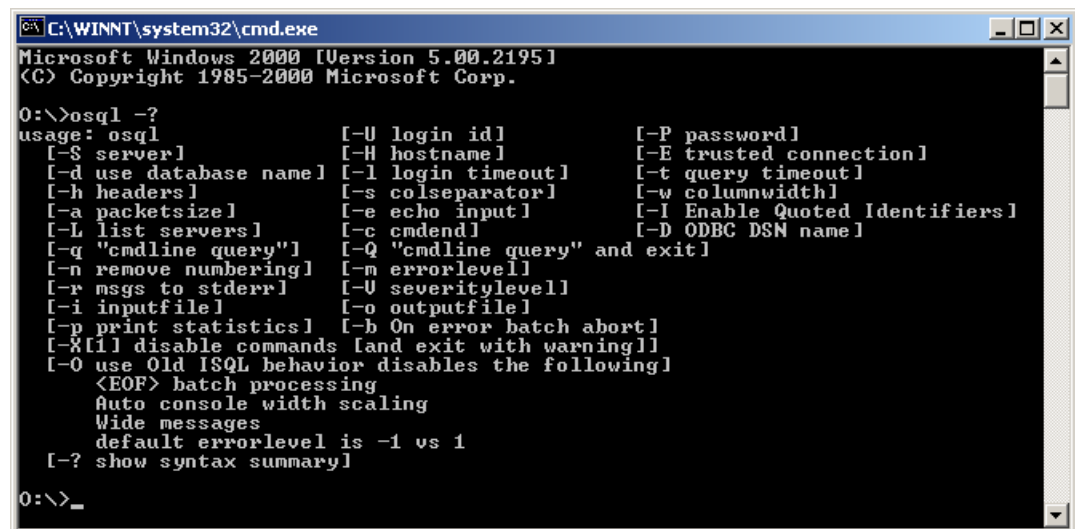
The following topics are covered:

- Logging into the OSQL Utility
- Creating a New MSDE SmarTeam Database
- Backing up and Restoring an MSDE Database
- Transporting a MSDE Database
- Renaming an MSDE database

Logging into the OSQL Utility

Displaying OSQL Help

To display OSQL Help type “OSQL /?” in the Windows command window (CMD):



```

C:\WINNT\system32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

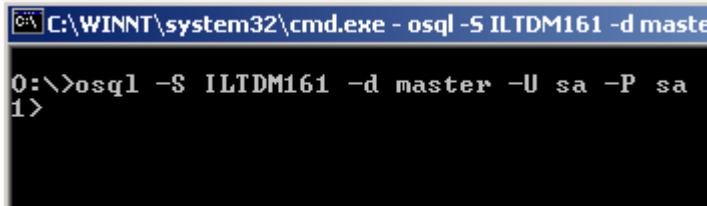
0:\>osql -?
usage: osql [-U login id] [-P password]
           [-S server] [-H hostname] [-E trusted connection]
           [-d use database name] [-l login timeout] [-t query timeout]
           [-h headers] [-s colseparator] [-w columnwidth]
           [-a packetsize] [-e echo input] [-i Enable Quoted Identifiers]
           [-L list servers] [-c cmdend] [-D ODBC DSN name]
           [-q "cmdline query"] [-Q "cmdline query" and exit]
           [-n remove numbering] [-m errorlevel]
           [-r msgs to stderr] [-U severitylevel]
           [-i inputfile] [-o outputfile]
           [-p print statistics] [-b On error batch abort]
           [-X[1] disable commands [and exit with warning]]
           [-O use Old ISQL behavior disables the following]
           <EOF> batch processing
           Auto console width scaling
           Wide messages
           default errorlevel is -1 vs 1
           [-? show syntax summary]

0:\>_

```

Logging into OSQL

To log in to OSQL enter: `OSQL -S server_name -d master -U login -P password`

Example

```
C:\WINNT\system32\cmd.exe - osql -S ILTDM161 -d master
O:\>osql -S ILTDM161 -d master -U sa -P sa
1>
```

You can now enter OSQL commands at the numbered prompts.

Note: In general you need to enter GO at the end of each OSQL command for the command to execute (See the examples in the sections below.)

Note: In the following examples, we assume that all operations are preceded by the login operation described in this section.

Creating a New MSDE SmarTeam Database

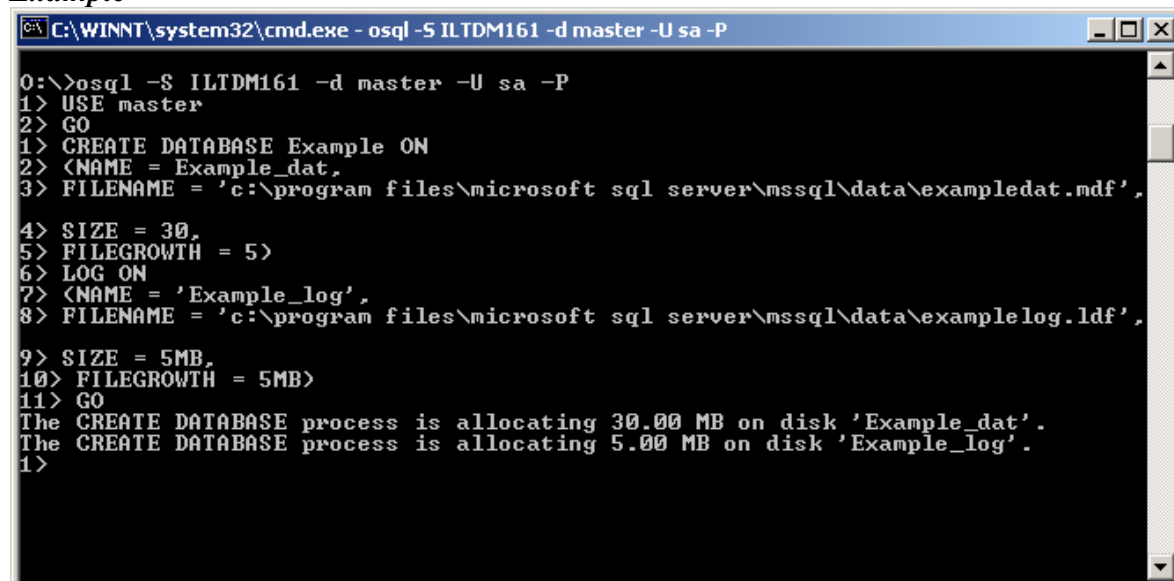
Creating a new MSDE SmarTeam database consists of two steps:

- Creating an empty MSDE database
- Creating a SmarTeam Database in the MSDE database

Creating an Empty MSDE database:

In the OSQL utility (command prompt) enter the following commands, which include the necessary parameters. The example below uses parameters relevant for the standard SmDemo database:

```
USE master
GO
CREATE DATABASE database_name ON
(NAME = Exampledat,
    FILENAME = 'c:\program files\microsoft sql
server\mssql\data\Exampledat.mdf',
    SIZE = 30,
    FILEGROWTH = 5)
LOG ON
(NAME = 'Examplelog',
    FILENAME = 'c:\program files\microsoft sql
server\mssql\data\Examplelog.ldf',
    SIZE = 5MB,
    FILEGROWTH = 5MB)
GO
```

Example


```

C:\WINNT\system32\cmd.exe - osql -S ILTDM161 -d master -U sa -P

0:\>osql -S ILTDM161 -d master -U sa -P
1> USE master
2> GO
1> CREATE DATABASE Example ON
2> (NAME = Example_dat,
3> FILENAME = 'c:\program files\microsoft sql server\mssql\data\exampledat.mdf',
4> SIZE = 30,
5> FILEGROWTH = 5)
6> LOG ON
7> (NAME = 'Example_log',
8> FILENAME = 'c:\program files\microsoft sql server\mssql\data\examplelog.ldf',
9> SIZE = 5MB,
10> FILEGROWTH = 5MB)
11> GO
The CREATE DATABASE process is allocating 30.00 MB on disk 'Example_dat'.
The CREATE DATABASE process is allocating 5.00 MB on disk 'Example_log'.
1>

```

Creating a SmarTeam Database in MSDE

Create a new SmarTeam database in MSDE using the Data Model Designer utility or SmartDBExplorer utility. Use the “sa” user to connect the database.

It is recommended to use the “sa” user as the SmarTeam database user. Refer to the Security section below for details.

Backing Up and Restoring an MSDE Database**Introduction**

This section shows how to use the command-line OSQL Utility to perform MSDE database backup and restore operations.

The section covers only the simple situation in which a MSDE database is backed up, becomes corrupted and is restored from the latest backup. It does not cover the case in which the user can choose between multiple backups of the database for restoring the system.

The following topics are presented:

- Preparing a backup device
- Backing up the database
- Restoring the database from backup
- Restoring an MSDE database with a different name

Preparing a Backup Device

To prepare backup device for future database backups issue the command below.

```
EXEC sp_addumpdevice 'disk', 'BackupName', DISK = 'Backup full path'
```

Where:

- The *server_name* is the name of the MSDE server you are connecting to, for example TDM325 or TDM325\SmarTeam

- The **login** is the login to the MSDE Server. The user logging in should be a DBA user, for example SA.
- **Password** -- Password means the password of the user logging in.
- **BackupName** – Logical name for backup
- **Backup full path** The Path to backup file location

Example

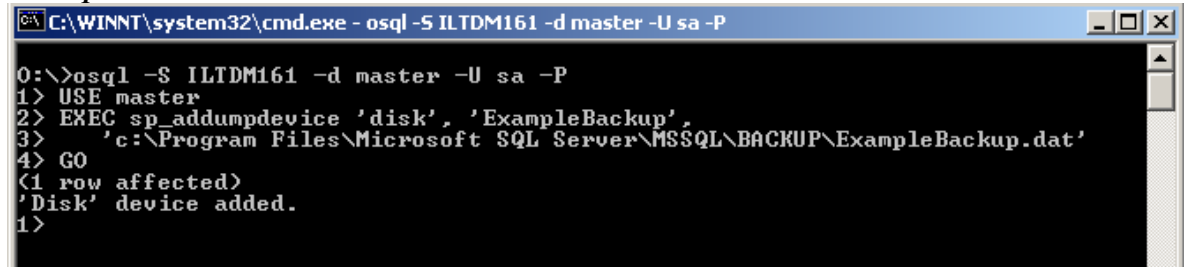
Here is an example how to prepare a backup device for SmDemo database.

```
EXEC sp_addumpdevice 'disk', 'SmDemoBackup', DISK
='c:\Program Files\Microsoft SQL
Server\MSSQL\BACKUP\SmDemoBackup.dat'
```

Important notes

- This operation should be done only **once** per each database you are going to backup.
- Please retain parameter designators exactly as they are written in the document, for example write **-S** and not **-s** for the server parameter.

Example



```
C:\WINNT\system32\cmd.exe - osql -S ILTDM161 -d master -U sa -P

0:\>osql -S ILTDM161 -d master -U sa -P
1> USE master
2> EXEC sp_addumpdevice 'disk', 'ExampleBackup',
3> 'c:\Program Files\Microsoft SQL Server\MSSQL\BACKUP\ExampleBackup.dat'
4> GO
<1 row affected>
'Disk' device added.
1>
```

Backing up the Database

To backup database, issue the command below:

```
BACKUP DATABASE Database_name TO BackupName
```

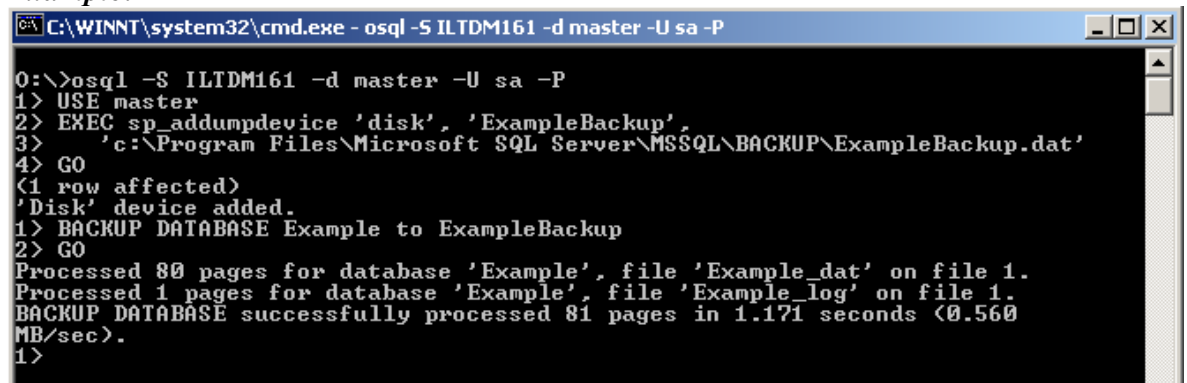
Where:

- **Database_name** – the name of the MSDE database you are backing up
- **BackupName** – the logical name for backup, as assigned in the previous step.

Example:

```
BACKUP DATABASE SmDemo TO SmDemoBackup
```

Example:



```
C:\WINNT\system32\cmd.exe - osql -S ILTDM161 -d master -U sa -P

0:\>osql -S ILTDM161 -d master -U sa -P
1> USE master
2> EXEC sp_addumpdevice 'disk', 'ExampleBackup',
3> 'c:\Program Files\Microsoft SQL Server\MSSQL\BACKUP\ExampleBackup.dat'
4> GO
<1 row affected>
'Disk' device added.
1> BACKUP DATABASE Example to ExampleBackup
2> GO
Processed 80 pages for database 'Example', file 'Example_dat' on file 1.
Processed 1 pages for database 'Example', file 'Example_log' on file 1.
BACKUP DATABASE successfully processed 81 pages in 1.171 seconds (0.560
MB/sec).
1>
```

Restoring the Database from Backup

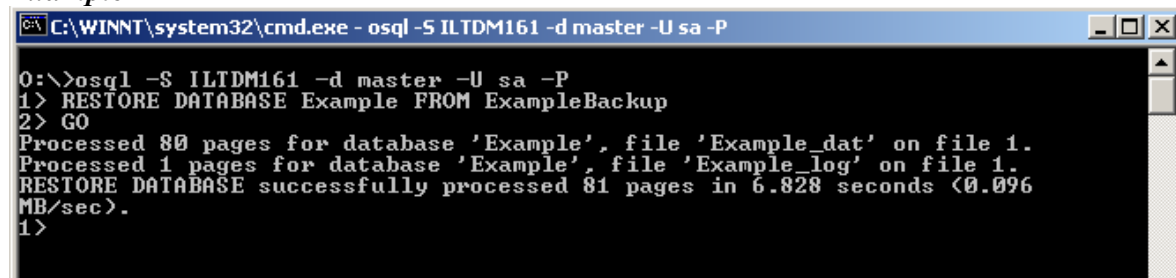
To restore database from backup, issue the following command:

```
RESTORE DATABASE Database_name FROM BackupName
```

Example:

```
RESTORE DATABASE SmDemo FROM SmDemoBackup
```

Example



```
C:\WINNT\system32\cmd.exe - osql -S ILTDM161 -d master -U sa -P
O:\>osql -S ILTDM161 -d master -U sa -P
1> RESTORE DATABASE Example FROM ExampleBackup
2> GO
Processed 80 pages for database 'Example', file 'Example_dat' on file 1.
Processed 1 pages for database 'Example', file 'Example_log' on file 1.
RESTORE DATABASE successfully processed 81 pages in 6.828 seconds (0.096
MB/sec).
1>
```

Restoring an MSDE Database with a Different Name

To restore a database from backup with a different name, proceed as follows:

```
USE master
GO
-- First determine the number and names of the files in the
backup.
-- MyNwind_2 is the name of the backup device.
RESTORE FILELISTONLY
    FROM ExampleBackup
-- Restore the files for MyNwind2_Test.
RESTORE DATABASE Example
    FROM ExampleBackup
    WITH RECOVERY,
    MOVE 'Exempladata' TO 'C:\Program Files\Microsoft SQL
Server\MSSQL\Data\exempladat.mdf',
    MOVE 'Examplelog' TO 'C:\Program Files\Microsoft SQL
Server\MSSQL\Data\examplelog.ldf'
GO
```

Transporting a MSDE Database

Occasionally, a SmarTeam administrator or implementer needs to send his database to other machines. This section explains how to do that safely. For the operations of this section the SmarTeam Database Connection Manager is available and can be used instead of OSQL; the use of each tool is described.

See Chapter 2 for an explanation of detaching and attaching a database and connecting a SmarTeam application to a database.

Transporting a MSDE Database includes the steps:

- Detach the MSDE database
- Send the data file and the transaction log file
- Attach the MSDE database and provide a connection string for a SmarTeam application

Detaching an MSDE Database File

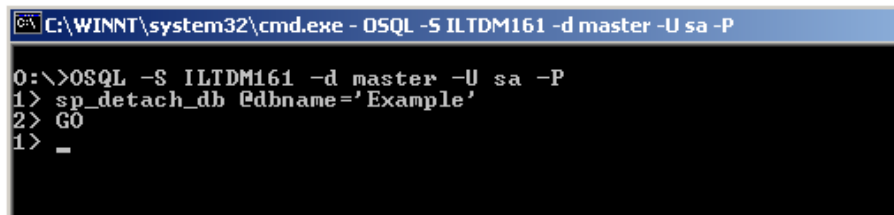
Normally, you detach a SmarTeam MSDE database from the MSDE server using Database Connection Manager.

Alternatively, you can use OSQL to detach the database using the following syntax:

```
sp_detach_db @dbname='database_name'
```

Example

```
sp_detach_db @dbname='Example'
```

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\WINNT\system32\cmd.exe' and the command 'OSQL -S ILTDM161 -d master -U sa -P'. The command prompt shows the following text:
O:\>OSQL -S ILTDM161 -d master -U sa -P
1> sp_detach_db @dbname='Example'
2> GO
1> -

Note: In case an error message is displayed that a user cannot be created, see the Troubleshooting chapter.

Attaching an MSDE Database File

This section describes how to attach a SmarTeam MSDE database to the MSDE server and to provide an open connection that can be utilized by a SmarTeam application to connect to the SmarTeam MSDE database.

In order to work with a database file, you must first “attach” it to the MSDE server. When you perform an Attach operation, you create a logical name for the database file. Once the file is attached, you can use the logical name to connect an application to the database.

Two methods are described for attaching a database:

- Using the SmarTeam Database Connection Manager
- Using the OSQL facility

Attaching a Database using the SmarTeam Database Connection Manager

To attach and connect to an MSDE database file using the SmarTeam Database Connection Manager:

1. From the Database Provider window, select **Attach a new MSDE database file to the server and connect to it** to attach a detached MSDE database to the database server and to create a Connection String for SmarTeam Applications. The Database Server window appears.
2. Select the server on which the new database will reside and enter the database administrator login name and password and click **Next**. The New Database Details window appears.
3. Select the MSDE file you want to attach and enter a name for the newly created database.

Note: If you select the name of an existing database, or a file that is already used by the database server, you are prompted to disconnect it and use the newly created one in its place. This may occur if the same database file or a different database file with the same name has already been attached. In such a case, investigate the reason behind the occurrence before proceeding.

4. Click **Next**.

The database file is attached to the database server. This may take a few moments.

5. Once the process is finished, click **Next** again.
6. Upon attaching the database, the SmarTeam Database Connection Manager creates a Connection String automatically.

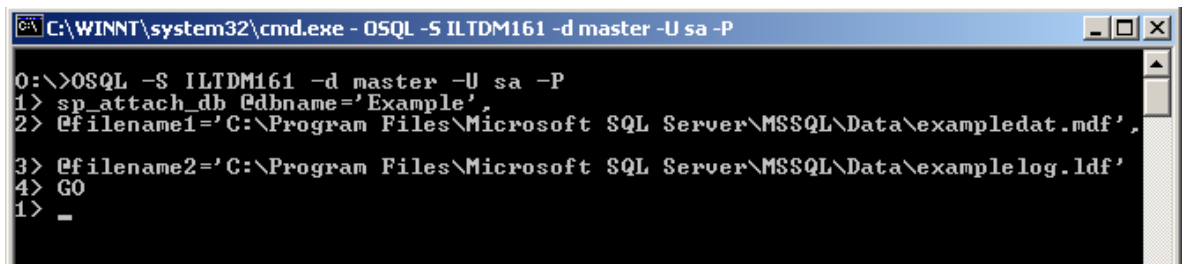
Attaching using the OSQL Facility

Use following command to attach an MSDE database using OSQL:

```
sp_attach_db [ @dbname = ] 'dbname'
            , [ @filename1 = ] 'filename_n' [ ,...16
```

Example

```
sp_attach_db @dbname='Example' ,
@filename1='C:\Program Files\Microsoft SQL
Server\MSSQL\Data\exampledat.mdf'
@filename2='C:\Program Files\Microsoft SQL
Server\MSSQL\Data\examplelog.ldf'
```



The screenshot shows a Windows command prompt window titled "C:\WINNT\system32\cmd.exe - OSQL -S ILTDM161 -d master -U sa -P". The command prompt displays the following text:

```
0:\>OSQL -S ILTDM161 -d master -U sa -P
1> sp_attach_db @dbname='Example' ,
2> @filename1='C:\Program Files\Microsoft SQL Server\MSSQL\Data\exampledat.mdf' ,
3> @filename2='C:\Program Files\Microsoft SQL Server\MSSQL\Data\examplelog.ldf'
4> GO
1> _
```

After attaching the database to the server, use the Database Connection Manager to create a SmarTeam connection to the database (see Appendix A for detailed information).

Renaming an Attached MSDE Database

If you want to rename an attached MSDE database use following command:

```
sp_renamedb [ @dbname = ] 'old_name' , [ @newname = ]
'new_name'
```

Example:

```
sp_renamedb @dbname = 'Example' , @newname = 'NewExample'
```

Resolving Names Duplication when Attaching MSDE Databases

There are situations when you need to attach a new database with the same identification parameters, such as the MSDE database name and/or SmarTeam database name/database ID, as one already existing in the MSDE Instance.

The following duplications are not allowed:

- Same database name for two different MSDE databases in an MSDE Instance
- Same SmarTeam database name/database ID for two different SmarTeam databases in an MSDE Instance, even though they are in different MSDE databases

Resolution of the following three scenarios are discussed:

1. Attaching an MSDE database to an MSDE Instance where an identical MSDE database name already exists in the MSDE Instance

2. Attaching an MSDE database to an MSDE Instance where an identical SmarTeam database name/database Id already exists for an MSDE database in the MSDE Instance
3. Attaching an MSDE database where the same MSDE database already exists, that is, the same MSDE name and the same SmarTeam name/database Id exist

1. An Identical MSDE Database Name Exists

If you want to attach a new MSDE database to an MSDE Instance where an MSDE database with the same name as the one you want to attach already exists, you need to perform one of the following actions:

- Rename the existing MSDE database inside the MSDE Instance using the OSQL `sp_renamedb` command as described in the relevant section, or use DBRegistration.
- Rename the new MSDE database as you attach it to the MSDE Instance using the OSQL command `sp_attach_db`, as described in the relevant section.

Note: If the files of the new MSDE database were originally located in the logical same path as the existing database, you would have to designate a different location for them.

If you do not perform either rename operation, the attach operation will fail.

2. An Identical SmarTeam Database Name/Database Id Exists

The Database Connection Manager will not attach a new MSDE database that contains a SmarTeam database with the same name and the same SmarTeam database ID as one already existing in the MSDE Instance.

First use the DBRegistration utility to re-register the existing SmarTeam database, obtaining a new name and Id, and then perform the attach operation using the OSQL command `sp_attach_db`.

3. The same MSDE database exists – same MSDE name and SmarTeam Name/Database Id

You should use a combination of the previous methods to insure that the names of the MSDE database and SmarTeam database would be unique inside the instance.

Security

User Management

You can use the “sa” user to access an MSDE databases. You set up the “sa” user during ST-standalone MSDE installation.

Changing an MSDE Database User Password

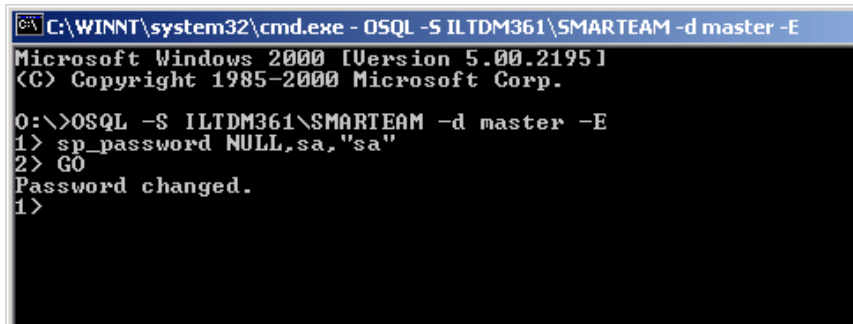
When the MSDE administrative password has been forgotten or misplaced, the following procedure can be used to set a new password for the administrator.

The procedure assumes an MSDE is installed with mixed Windows/MSDE authentication model, which is the SmarTeam default.

Log in to the OSQL utility using trusted authentication mode (-E option). You should be logged on in Windows as a local computer administrator.

To log in to OSQL enter:

```
OSQL -S server_name -d master -E
sp_password NULL, <newpassword>, "sa"
GO
```

A screenshot of a Windows command prompt window. The title bar reads "C:\WINNT\system32\cmd.exe - OSQL -S ILTDM361\SMARTTEAM -d master -E". The window content shows the following text:

```
Microsoft Windows 2000 [Version 5.00.2195]  
<C> Copyright 1985-2000 Microsoft Corp.  
  
O:\>OSQL -S ILTDM361\SMARTTEAM -d master -E  
1> sp_password NULL,sa,"sa"  
2> GO  
Password changed.  
1>
```


Chapter 4: Troubleshooting

This section discusses common issues and their solutions.

Issue: Attach fails

Problem: When you try to Attach an MSDE Database, the error message "... Cannot locate fileLDF" appears.

Cause: When running the application over the database (whether it is the Data Model Designer on WizSrc or the regular SmarTeam on SmDemo), transactions are created and REDO information is written into the transaction log (LDF file). The problem happens when a database is detached and an MDF file is sent to the user without including the LDF file. When you try to attach the MDF file, the error occurs, as MSDE requires REDO data to handle a database correctly.

Solution: Request the customer to detach a database properly and send you both database and transaction log files.

Issue: SmarTeam Connect fails

Problem: You cannot establish connection to SmarTeam either in the Database Connection Manager or in SmarTeam itself. This may originate from one of several sources.

Security – Windows-based authentication

When you try to connect, the error message "... failed ... no trust relationship..." appears.

Cause: SmarTeam uses SQL Server authentication (login *smarteam* has been created in Security Section of Enterprise Manager), while the SQL Server current authentication setting is Windows Authentication only.

Solution: Open the Enterprise Manager, right click on the server, Security tab and change the authentication mode to "Windows and SQL Server authentication". After restarting the SQL Server instance (usually occurs automatically), the problem is resolved.

SmarTeam cannot locate its tables

When you try to connect, the error message "... TDM_DB_VERSION cannot be found..." appears. This may have one of two causes.

Cause 1: Privileges have not been granted to the user you are using

Solution:

1. Open the Enterprise Manager, Security Section, and open your login script in the edit window (e.g., *smarteam*).
2. In the General tab, set the default database to the database to which you want to connect (e.g. *SmDemo*)
3. In the Database Access tab, check "Permit" for that database in the upper list and choose "public" and "db_owner" roles in the bottom list.
4. Press Okay to close the edit window.

Cause 2: The Connect operation was not performed by the owner (orphan user, tables do not belong to dbo)

Cause: A database was received from a customer (QA, R&D) and they have built the SmarTeam data structure by a database user (e.g. *customer*), who is not

db_owner. In this case, you receive that database, and you do not have the login *customer* on your local server. You cannot create that login, as the SQL Server creates the login and user together and the user *customer* already exists in the database *SmDemo* you received.

Solution: The simplest solution is to change the owner of all SmarTeam tables to the **dbo** database user. You can do this by running two scripts in Query Analyzer.

1. Run following script in Query Analyzer as **sa** (server administrator) user.

```
declare @U sysname
```

```
declare @UID smallint
```

```
SET @U='customer' -- Customer's user name parameter - Change it to customer's database SmarTeam tables user-owner name
```

```
USE CustomerDatabase -- Database name parameter – Change it to your database name
```

```
SET @UID=(select uid from sysusers where name=@U)
```

```
select 'exec sp_changeobjectowner @objname=' + @U + '.' + name + ',  
@newowner="dbo" + ';' from sysobjects where uid=@UID order by name
```

Note: The user name '*customer*' is case-sensitive.

This script dynamically creates another script, that reflects the exact structure of your database and its output is actually another script, the real one you must use to transfer ownership.

2. The previous script will produce output. Right click in the **Output** sub-window, choose **Select All**, then copy and paste it to the new Query Analyzer window. Run it. This operation transfers ownership to the user **dbo**.
3. Remove the old user using the following command:

```
EXEC sp_dropuser @name_in_db = 'user'
```

For Example:

```
EXEC sp_dropuser @name_in_db = 'customer'
```

4. Run the Database Connection Manager to attach the database.

Appendix A: Database Connection

This appendix describes how to use the Database Connection Manager to create a connection string for connecting a SmarTeam application to a MSDE database.

1. Launch the Database Connection Manager:

Select Start → Programs → SmarTeam → Administrative Tools → Database Connection Manager.

The SmarTeam Database Connection Manager window appears, displaying currently connected databases.

2. In the **Connection tab page, complete the following fields:**

- ☐ Server
- ☐ Username
- ☐ Password
- ☐ Database Name

3. Click **Test Connection to test the connection from SmarTeam to the MSDE database.**

Note: This option applies to connecting to databases that are already attached to the database server. Be sure to choose the **Select the database on the server** and *not* the **Attach a database file as a database name** radio button at this point.

4. After connection is confirmed, the connection process is complete. Click **OK to return to the **Database Connection String** window.**

In the **Database Connection String** window, the connection string is displayed in the **Connection String** field.

