TelePresence

Migrating TelePresence Management Suite (TMS) to a New Server

Migrating TelePresence Management Suite (TMS) to a New Server
September 2012 Edition

Table of Contents

List of Tables

1 Introduction

This document describes procedures for migrating a TelePresence Management Suite (TMS) database application. The details of this document will provide detailed instructions on moving TMS from one server to another, with the the options of moving the location of the SQL database. Keep in mind that there are no methods of migrating local user accounts form one Windows server to another. If you are using local Windows accounts for access to the TMS server, you will need to manually create these accounts on the new server.

1.1 Release Notes

Table 1 - Release Notes

Technical Change	Title(s) of Affected Section(s)	Changes Made By	Date
Initial release.		Zac Colton	7/17/2012
Revised title			8/9/2012
Revised sections	Restoring SQL Database and Using TMS Provisioning Extension	Zac Colton	9/28/2012
Revised sections	Added SQL Management Studio instructions and updated TMSAgent/TMSPE sections	Vernon Depee	2/12/2012

2 Migrating the TMS

2.1 Prerequisites

If using TMS Legacy Agents, be sure to first disable replication to all devices.

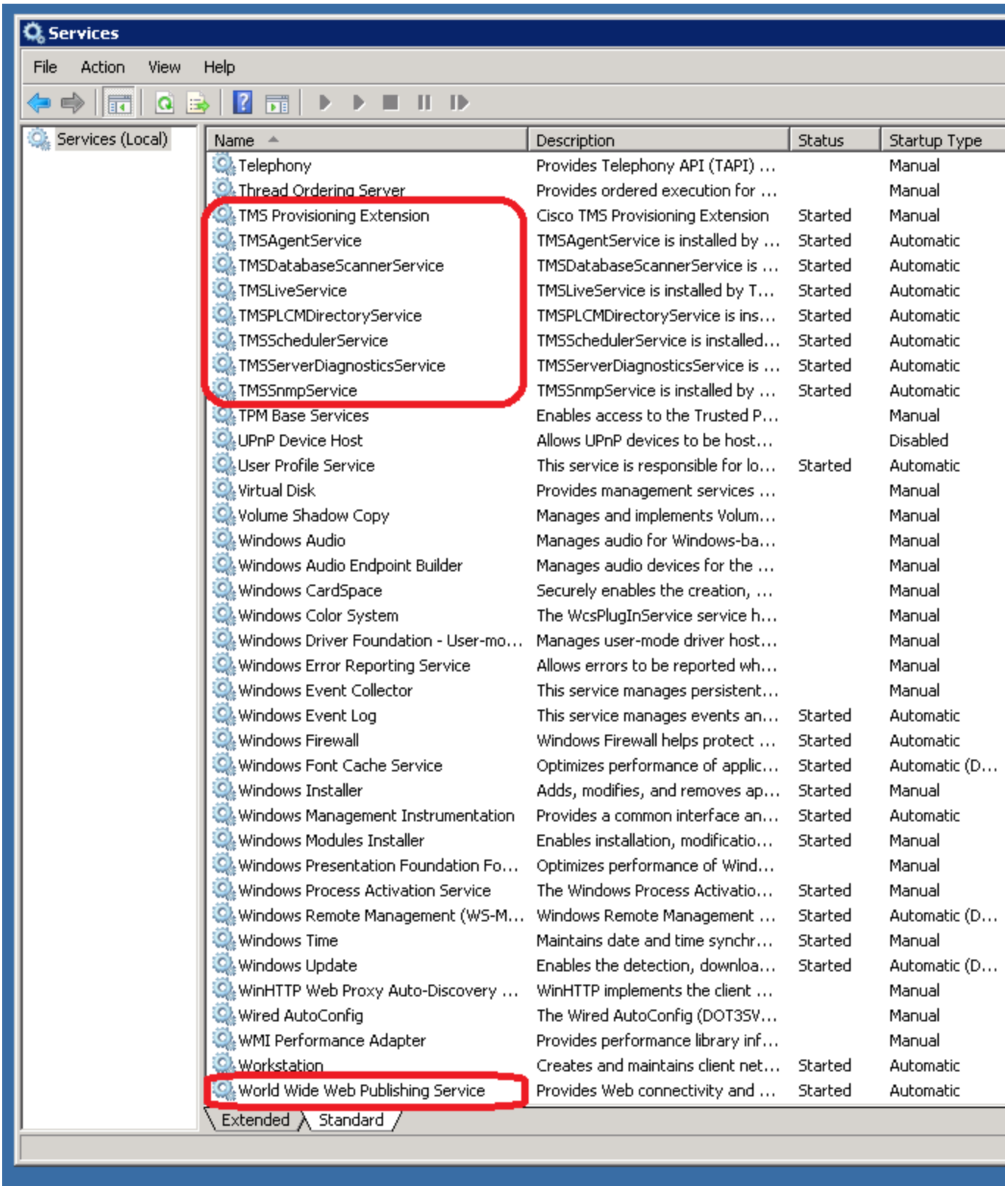
Shut down all TMS Windows services:

- TMSAgentService
- TMSDatabaseScannerService
- TMSLiveService
- TMSPLCMDirectoryService
- TMSSchedulerService
- TMSServerDiagnosticsService
- TMSSNMPService

If using TMS Provisioning Extension, also shut down:

- TMS Provisioning Extension

Note: If using TMS Legacy Agents, the FQDN configured at the bottom of the TMSAgent settings page of the TMS (typically the FQDN of the TMS as configured in Windows) must be resolvable to the ip address of the TMS when the VCS does a lookup on this FQDN. Reverse lookups must also be present.



2.2 Backup SQL Database

There are two methods of backing up and restoring the SQL database. The CLI is available on all systems that have SQL installed, but SQL management studio can be downloaded from Microsoft and used if desired.

2.2.1 Using SQL CLI (Recommended)

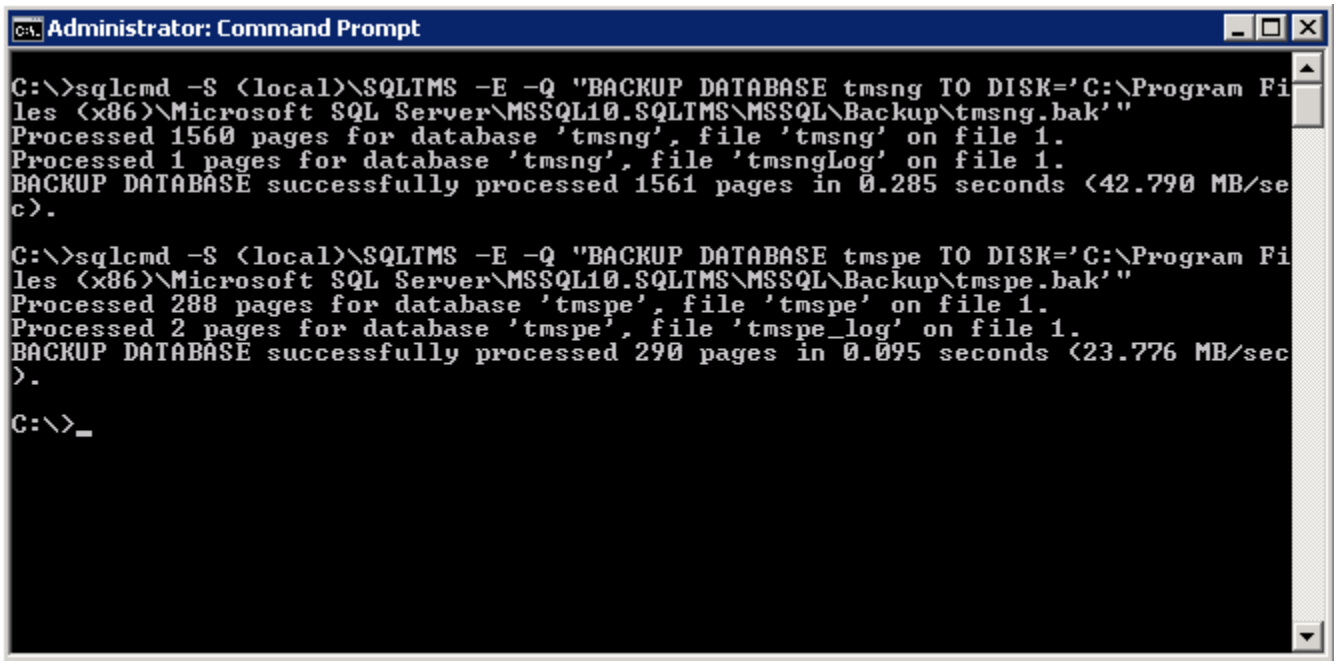
If the SQL database currently resides on the same server as the TMS server application, and you plan on moving the database to a new server, either the new Windows server hosting TMS or a seperate SQL instance, the database needs to be backed up and restored to the new location.

The following commands are run within command prompt on the current TMS server that hosts the SQL Express instance with the tmsng database. These command are written to allow SQL server access using the currently logged on Windows user. To use SQL login credentials, replace "-E" with "-U <username> -P <password>" and replace the username and password with SQL credentials with system admin rights.

2.2.1.1 TMS Database

1. To backup the tmsng SQL database from command line: (Replace <path> with the location you would like to save the backup to. This location needs to have adequate space for the backup, and the SQL service needs to have access to it.

   sqlcmd -S (local)\SQLTMS -E -Q "BACKUP DATABASE tmsng TO DISK=***'<path>***tmsng.bak'"



2.2.1.2 TMSPE Database (Only if using TMSPE)

2. If using TMS Provisioning Extension, to backup the tmspe SQL database from command line:

   sqlcmd -S (local)\SQLTMS -E -Q "BACKUP DATABASE tmspe TO DISK='**<path>**tmspe.bak'

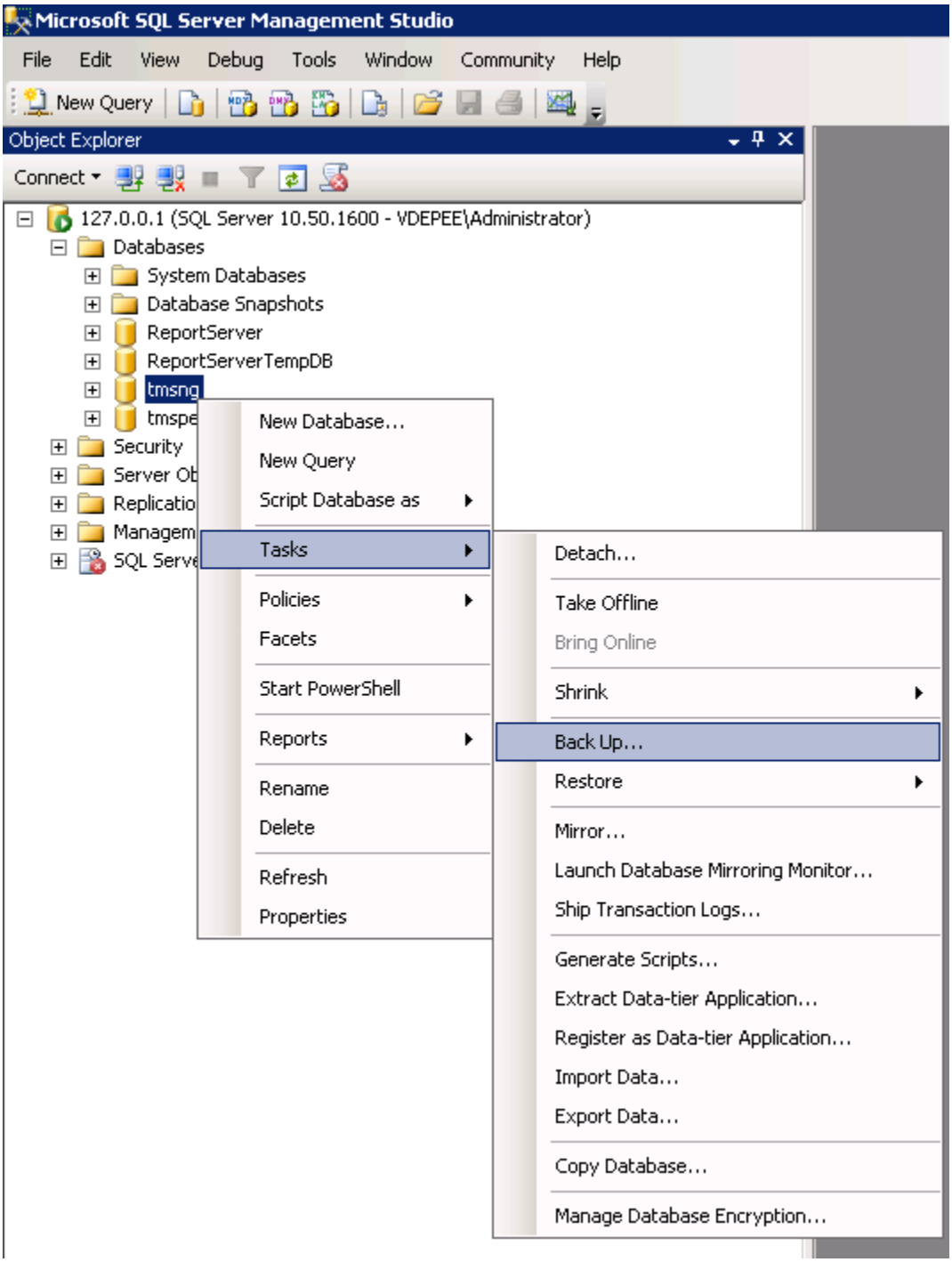
2.2.1.3 Copy Files to new Server

3. Copy the backup file(s) to the new SQL server location. This can either be a separate SQL server, or the new TMS server location that is running SQL Server Express. The location that you copy the backup file to requires the SQL service user to have full access.

2.2.2 Using SQL Management Studio (CLI Recommended)

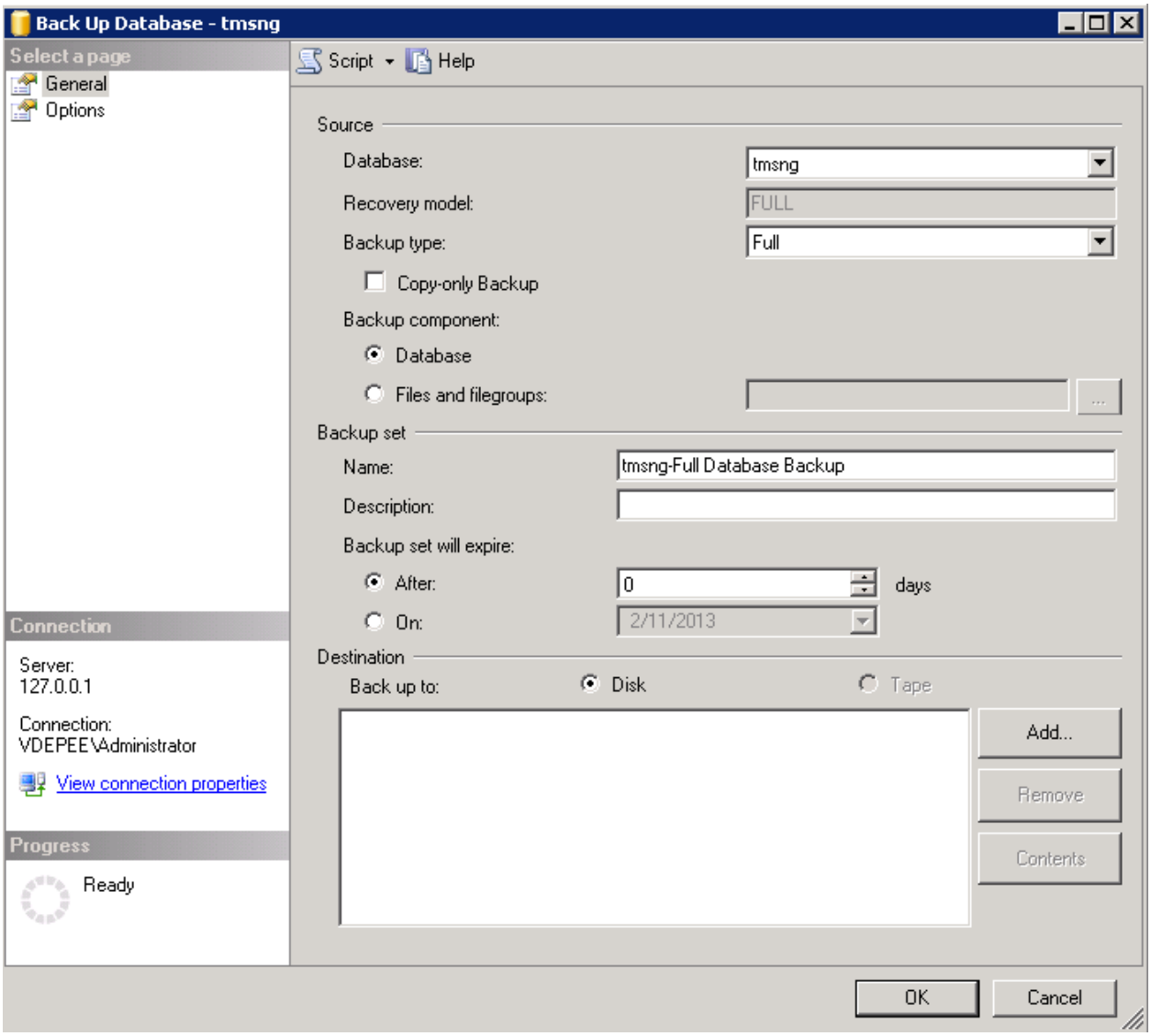
2.2.2.1 TMS Database

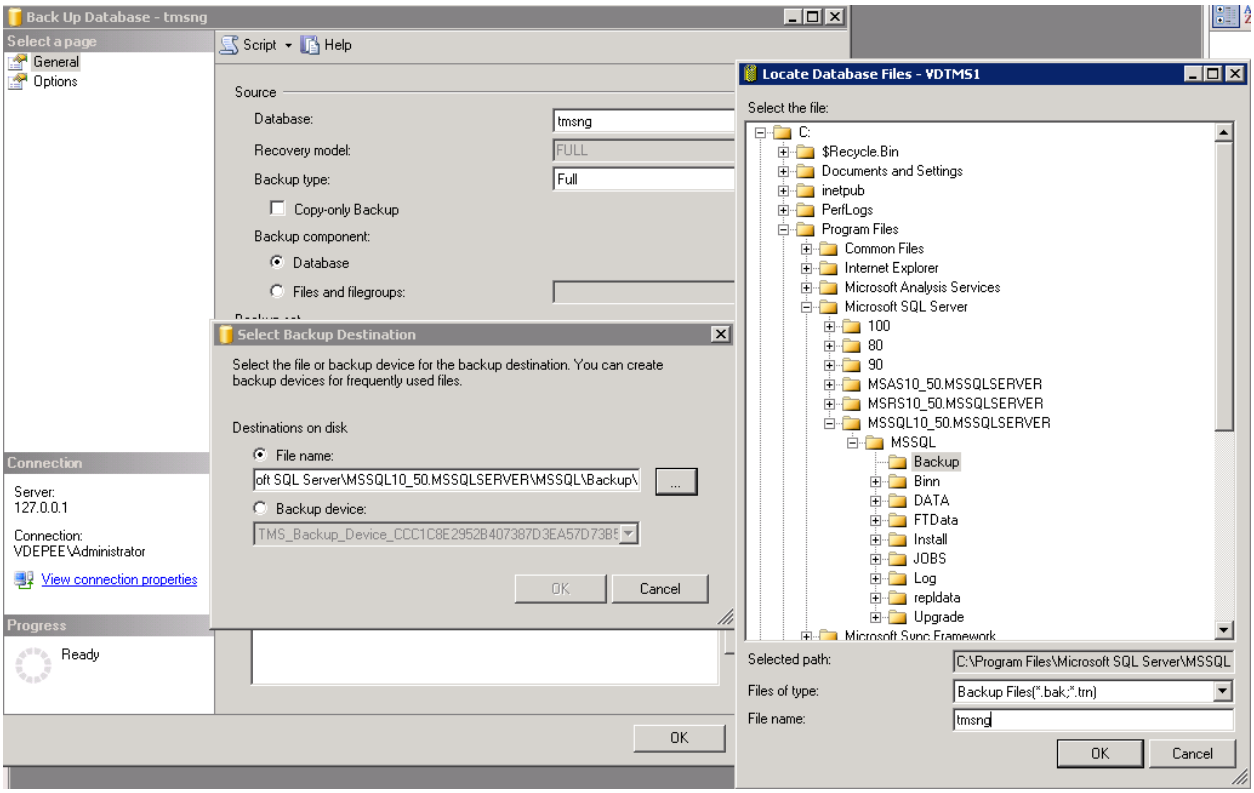
On the current SQL server, open up SQL management Studio and navigate to the tmsng database. Right click on the database and navigate to tasks > back up…



In the backup prompt page, make sure that the settings are as shown below. There may be a default destination specified. If you would like to use this location to store the backup, just go ahead and click ok

and the backup will be made in the specified location. If there is no destination location specified, select Add and click on … and enter the file name of tmsng. Then select OK on all three screens and the database backup should occur.





Note: If you get an access denied message back, make sure that you are writing to a location that the SQL service user can write to. I recommend using the Backup folder inside of MSSQL as you should be able to write successfuly there.

2.2.2.2 TMSPE Database (Only if using TMSPE)

If you are using TMSPE, follow the steps as above again, but this time right-click on the tmspe database instead of the tmsng database. Name this packup tmspe.bak.

2.2.2.3 Copy Files to new Server

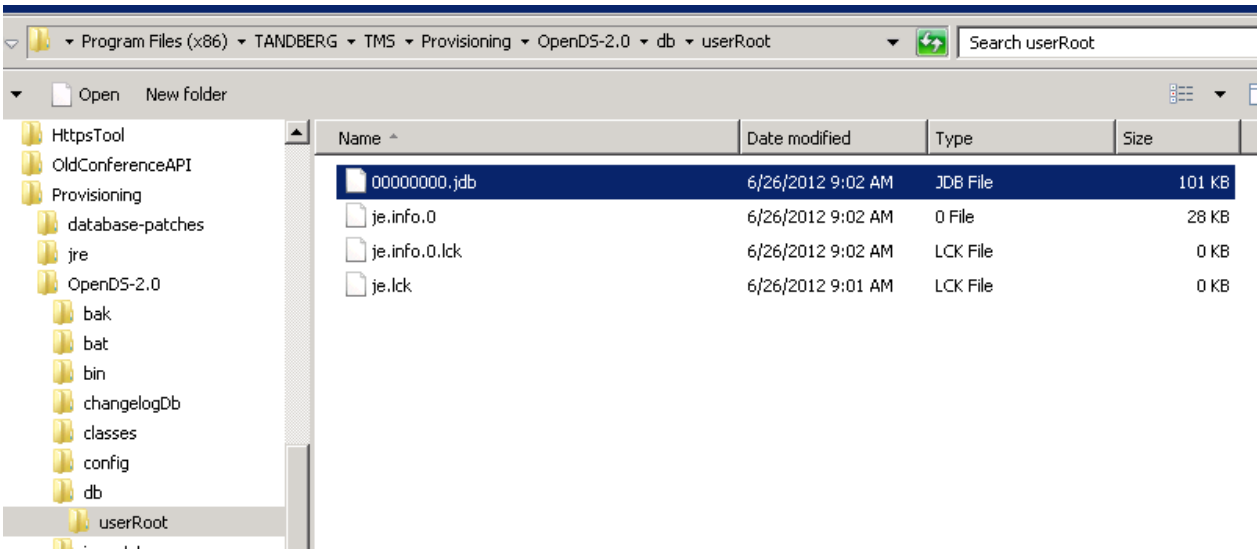
Copy the backup file(s) to the new SQL server location. This can either be a separate SQL server, or the new TMS server location that is running SQL Server Express. The location that you copy the backup file to requires the SQL service user to have full access.

2.3 If using TMS Legacy Agents

If using TMS Legacy Agents, on the old TMS server, open Windows Explorer and go to:

%OPENDS_HOME%\db\userRoot

Example 1: *C:\Program Files\TANDBERG\TMS\provisioning\OpenDS-2.0\db\userRoot*



1. Copy the *.jdb to a temporary location on the new TMS server

2.4 Other local files to be save

The TMS may also contain end-user customized files. These locations should be checked. If there are customizes files in these location, be sure to save these files and copy them to the new TMS server location after it has been reinstalled. The directories listed below are the the default paths.

C:\Program Files\TANDBERG\TMS\wwwTMS\Data\CiscoSettings
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\CompanyLogo
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\ExternalSourceFiles
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\Image
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\Language
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\Logo
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\Map
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\MGCSettings
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\Software
C:\Program Files\TANDBERG\TMS\wwwTMS\Data\Sound
C:\Program Files\TANDBERG\TMS\wwwTMS\Public\Data\SOFTWARE

2.5 Restoring SQL Database

There are two methods of backing up and restoring the SQL database. The CLI is available on all systems that have SQL installed, but SQL management studio can be downloaded from Microsoft and used if desired.

2.5.1 Using SQL CLI (Recommended)

2.5.1.1 TMS Database

A running copy of SQL Server (2005/2008) or SQL Server Express (2005/2008) is needed to continue. If you will be using SQL Express on the new TMS server to house the database, install TMS, and then uninstall TMS. This will allow the installer to create the SQL Express instance that will be needed to restore the database. During this install, you do not need to include the release key and option keys as this data is overwritten when the database is restored from backup. If you are using a separate SQL server, you do not need to install and uninstall TMS.

On the server, running either SQL Server or SQL Server Express, run the following command to restore the SQL database. These command are written to allow SQL server access using the currently logged on

Windows user. To use SQL login credentials, replace "-E" with "-U <username> -P <password>" and replace the username and password with SQL credentials with system admin rights.

1. Replace **servername** with the hostname of the SQL server.
2. Replace the **instancename** with the SQL service instance name.
3. Replace the <**pathofbackup**> variable with the location of the backup (.bak) file.
4. Replace the <**pathofdbfiles**> variable with the location that you wish to store the database mdf file (tmsng_data.mdf) and the location you wish to store the database ldf file (tmsng_log.ldf

If Restoring to SQL 2005:

sqlcmd -S **<servername\instancename>** -E -Q "DECLARE @Table TABLE (LogicalName varchar(128),[PhysicalName] varchar(128), [Type] varchar, [FileGroupName] varchar(128), [Size] varchar(128), [MaxSize] varchar(128), [FileId]varchar(128), [CreateLSN]varchar(128), [DropLSN]varchar(128), [UniqueId]varchar(128), [ReadOnlyLSN]varchar(128), [ReadWriteLSN]varchar(128), [BackupSizeInBytes]varchar(128), [SourceBlockSize]varchar(128), [FileGroupId]varchar(128), [LogGroupGUID]varchar(128), [DifferentialBaseLSN]varchar(128), [DifferentialBaseGUID]varchar(128), [IsReadOnly]varchar(128), [IsPresent]varchar(128));DECLARE @Path varchar(1000);SET @Path='**<pathofbackup>**\tmsng.bak';DECLARE @LogicalNameData varchar(256),@LogicalNameLog varchar(256);INSERT INTO @table EXEC('RESTORE FILELISTONLY FROM DISK="' +@Path+ '"');SET @LogicalNameData=(SELECT LogicalName FROM @Table WHERE Type='D');SET @LogicalNameLog=(SELECT LogicalName FROM @Table WHERE Type='L');RESTORE DATABASE tmsng FROM DISK='**<pathofbackup>**\tmsng.bak' WITH REPLACE, MOVE @LogicalNameData TO '**<pathofdbfiles>**\tmsng_data.mdf', MOVE @LogicalNameLog TO '**<pathofdbfiles>**\tmsng_log.ldf'"

If Restoring to SQL 2008:

sqlcmd -S **<servername\instancename>** -E -Q "DECLARE @Table TABLE (LogicalName varchar(128),[PhysicalName] varchar(128), [Type] varchar, [FileGroupName] varchar(128), [Size] varchar(128), [MaxSize] varchar(128), [FileId]varchar(128), [CreateLSN]varchar(128), [DropLSN]varchar(128), [UniqueId]varchar(128), [ReadOnlyLSN]varchar(128), [ReadWriteLSN]varchar(128), [BackupSizeInBytes]varchar(128), [SourceBlockSize]varchar(128), [FileGroupId]varchar(128), [LogGroupGUID]varchar(128), [DifferentialBaseLSN]varchar(128), [DifferentialBaseGUID]varchar(128), [IsReadOnly]varchar(128), [IsPresent]varchar(128), [TDEThumbprint]varchar(128));DECLARE @Path varchar(1000);SET @Path='**<pathofbackup>**\tmsng.bak';DECLARE @LogicalNameData varchar(256),@LogicalNameLog varchar(256);INSERT INTO @table EXEC('RESTORE FILELISTONLY FROM DISK="' +@Path+ '"');SET @LogicalNameData=(SELECT LogicalName FROM @Table WHERE Type='D');SET @LogicalNameLog=(SELECT LogicalName FROM @Table WHERE Type='L');RESTORE DATABASE tmsng FROM DISK='**<pathofbackup>**\tmsng.bak' WITH REPLACE, MOVE @LogicalNameData TO '**<pathofdbfiles>**\tmsng_data.mdf', MOVE @LogicalNameLog TO '**<pathofdbfiles>**\tmsng_log.ldf'"

2.5.1.2 TMSPE Databse (Only if using TMSPE)

To restore the tmspe SQL database from command line:

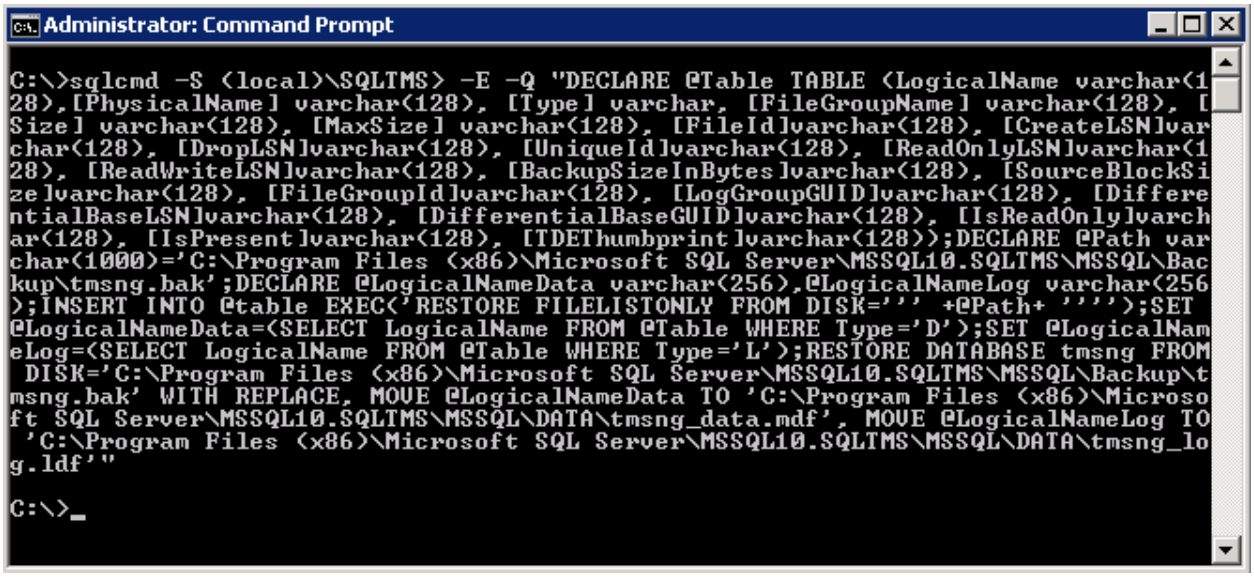
If Restoring to SQL 2005:

sqlcmd -S **<servername\instancename>** -E -Q "DECLARE @Table TABLE (LogicalName varchar(128),[PhysicalName] varchar(128), [Type] varchar, [FileGroupName] varchar(128), [Size] varchar(128), [MaxSize] varchar(128), [FileId]varchar(128), [CreateLSN]varchar(128), [DropLSN]varchar(128), [UniqueId]varchar(128), [ReadOnlyLSN]varchar(128), [ReadWriteLSN]varchar(128), [BackupSizeInBytes]varchar(128), [SourceBlockSize]varchar(128), [FileGroupId]varchar(128), [LogGroupGUID]varchar(128), [DifferentialBaseLSN]varchar(128), [DifferentialBaseGUID]varchar(128), [IsReadOnly]varchar(128), [IsPresent]varchar(128));DECLARE

@Path varchar(1000);SET @Path='**<pathofbackup>**\tmspe.bak';DECLARE @LogicalNameData varchar(256),@LogicalNameLog varchar(256);INSERT INTO @table EXEC('RESTORE FILELISTONLY FROM DISK="' +@Path+ '"');SET @LogicalNameData=(SELECT LogicalName FROM @Table WHERE Type='D');SET @LogicalNameLog=(SELECT LogicalName FROM @Table WHERE Type='L');RESTORE DATABASE tmspe FROM DISK='**<pathofbackup>**\tmspe.bak' WITH REPLACE, MOVE @LogicalNameData TO '**<pathofdbfiles>**\tmspe_data.mdf', MOVE @LogicalNameLog TO '**<pathofdbfiles>**\tmspe_log.ldf'"

If Restoring to SQL 2008:

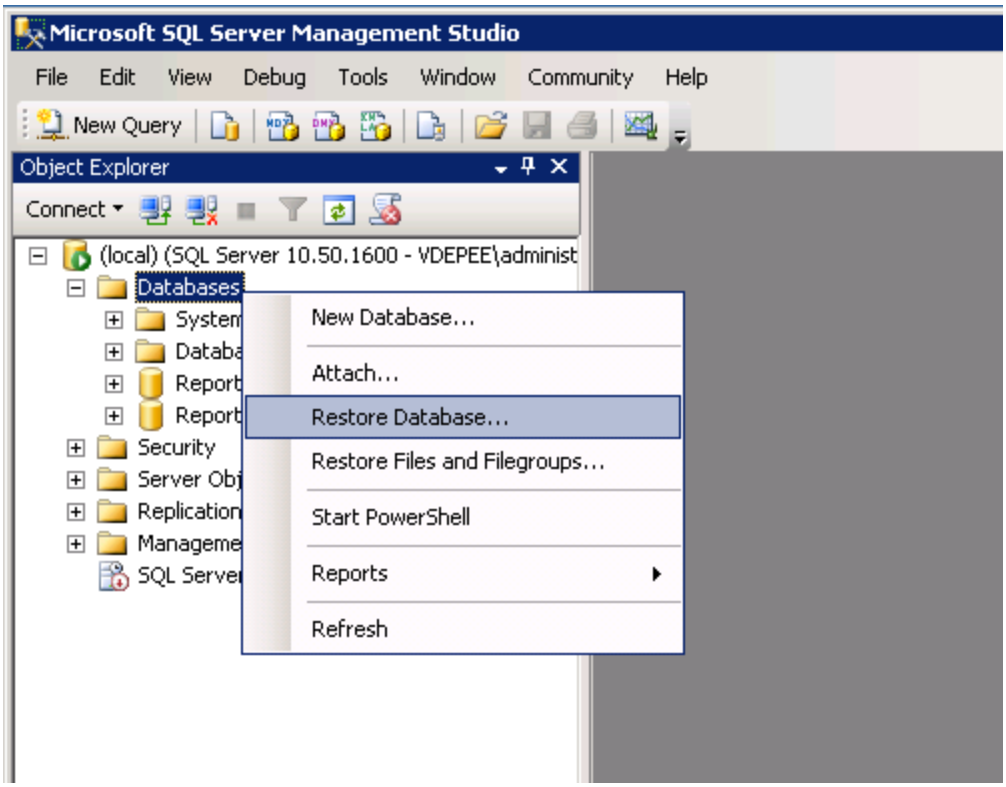
sqlcmd -S **<servername\instancename>** -E -Q "DECLARE @Table TABLE (LogicalName varchar(128),[PhysicalName] varchar(128), [Type] varchar, [FileGroupName] varchar(128), [Size] varchar(128), [MaxSize] varchar(128), [FileId]varchar(128), [CreateLSN]varchar(128), [DropLSN]varchar(128), [UniqueId]varchar(128), [ReadOnlyLSN]varchar(128), [ReadWriteLSN]varchar(128), [BackupSizeInBytes]varchar(128), [SourceBlockSize]varchar(128), [FileGroupId]varchar(128), [LogGroupGUID]varchar(128), [DifferentialBaseLSN]varchar(128), [DifferentialBaseGUID]varchar(128), [IsReadOnly]varchar(128), [IsPresent]varchar(128), [TDEThumbprint]varchar(128));DECLARE @Path varchar(1000);SET @Path='**<pathofbackup>**\tmspe.bak';DECLARE @LogicalNameData varchar(256),@LogicalNameLog varchar(256);INSERT INTO @table EXEC('RESTORE FILELISTONLY FROM DISK="' +@Path+ '");SET @LogicalNameData=(SELECT LogicalName FROM @Table WHERE Type='D');SET @LogicalNameLog=(SELECT LogicalName FROM @Table WHERE Type='L');RESTORE DATABASE tmspe FROM DISK='**<pathofbackup>**tmspe.bak' WITH REPLACE, MOVE @LogicalNameData TO '**<pathofdbfiles>**\tmspe_data.mdf', MOVE @LogicalNameLog TO '**<pathofdbfiles>**\tmspe_log.ldf'"



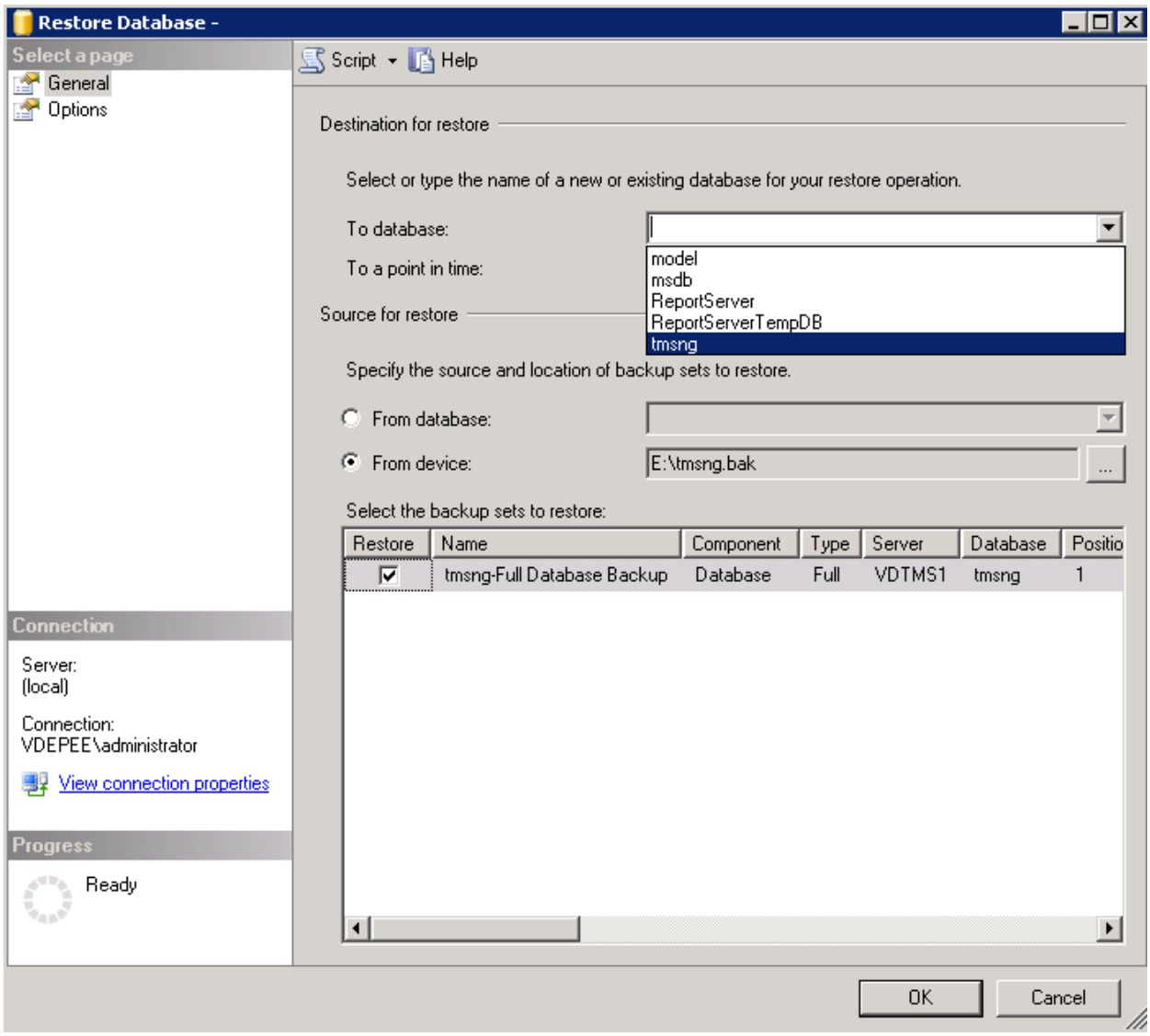
2.5.2 Using SQL Management Studio (CLI Recommended)

2.5.2.1 TMS Database

On the new SQL Server, open up SQL Management studio. Right click on Databases and select Restore Database.



Under Source for restore, select file and point to the tmsng.bak file. Then in the select the backup sets to restore field, check the tmsng backup. After checking this backup, on the To database: field on the top, tmsng should appear as an option. Select tmsng and click on OK. This should restore the tmsng database.



2.5.2.2 TMSPE Database (Only if using TMSPE)

The process for restoring the TMSPE database is the same as the TMSNG database, just select the tmspe backup file instead of the tmsng backup file and select the tmspe database instead of the tmsng database.

On the new database server, ensure that the SQL Browser service is running. If it is not running, the TMSPE installation will fail.

2.6 After Restoring the Database

After restoring the database:

1. Reinstall TMS on the new server to host the TMS server application.
2. Select "***custom install***" to point to the new SQL server location.

2.7 If using TMSPE

Reinstall TMSPE on the TMS server and point to the new database location.

2.8 If using TMS Legacy Agents

1. Stop the TMSAgents Windows service. This also stops the OpenDS Windows Service
2. Browse to %OPENDS_HOME%\db\userRoot

   Example 2: *C:\Program Files\TANDBERG\TMS\provisioning\OpenDS-2.0\db\userRoot*
3. Delete all of the files that exist in that folder.
4. Move the .jdb files copied from the old TMS server into that folder.
5. Start the TMSAgentService Windows Service

DO NOT YET ACCESS THE TMS PORTAL

2.9 Running TMS Tools Utility

If the the host name of the TMS server has changed AND you use local user acounts (user accounts that exist on the server that hosts the TMS server application - not Active Directory accounts), you will need to run the TMS Tools utility to modify data within the database to be sure you can login:

- TMS Tools > Utilities > Change Users Domain
- Old Domain Name: <old TMS server host name>
- New Domain: <new TMS server host name>

Failure to do this could result in the loss of the ability to access the TMS portal.

NOTE:

The local useraccounts are not moved to the the new server during this migration process. If you are using local Windows accounts, these accounts need to be manually recreated on the new Windows server that TMS is being migrated to.

3 Post Installation

To access the TMS Portal with a user account that has Site Administrator rights in TMS:

1. Go to **Administrative Tools** > **Configuration** > **General Settings**
2. Make sure that the value for the Software FTP Directory is accurate for the new server install. This value could be wrong if you have installed TMS on a different drive letter on the new server when compared to the old server, or if you switched from a 32-bit to a 64-bit version of the Windows Server OS.
3. Go to **Administrative Tools** > **Configuration** > **Network Settings**
4. Check the values for the following fields:

- General Network Settings > URL Where Software Packages Can Be Downloaded
- Advanced Network Settings for Systems on Internal LAN > TMS Server IPv4 Address
- Advanced Network Settings for Systems on Internal LAN > TMS Server IPv6 Address
- Advanced Network Settings for Systems on Internal LAN > TMS Server Fully Qualified Host Name
- Advanced Network Settings for Systems on Public Internet/Behind Firewall > TMS Server Address (Fully Qualified Host Name or IPv4 Address)

3.1 If using TMS Legacy Agents

When using TMS legacy agents:

1. Go to **Administrative Tools** > **Configuration** > **TMS Agent Settings**
2. Under **Global > Settings**, set a password for the **LDAP Configuration Password** and **LDAP Replication Password**. This will ensure that these passwords will be in sync for where this data is stored.

3. Make sure that the value for TMS Agent Backup > Backup Directory is accurate for the new server install. This value could be wrong if you have installed TMS on a different drive letter on the new server when compared to the old server, or if you switched from a 32-bit to a 64-bit version of the Windows Server OS.

3.1.1 Deleting Older Server

At the bottom of the page, under TMS Servers, you may now see multiple TMS servers. If so, delete the old server. To do so:

1. Click the machine name and select **Delete**.
2. For the new machine, verify that the Network Address is correct (the FQDN and the IP address).

If not correct:

1. Select **Edit** on the right-hand side.
2. Type in the correct FQDN address of the new TMS server and select update.
3. At the top of the page, type in a new LDAP Configuration Password and LDAP Replication Password under **Global** > **Settings**.
4. Select **Save** at the bottom of the page.

3.1.2 Running the TMS Agent Diagnostics

After these steps have been completed:

Run the TMS Agent Diagnostics on the Local TMS Agent by going to **Administrative Tools > TMS Agent Diagnostics**

NOTE: If the **TMS Agent Diagnostics** fail, please refer to the **Cisco TMS Agent Troubleshooting Procedures** guide.

3.1.3 Enabling the TMS Agent Replication to the VCS.

- If using TMS Provisioning Extension, TMSPE needs to be reinstalled and pointed to the current location of the tmspe database
- If using Analytics Extensions and it was installed on the old TMS server, it needs to be reinstalled to the new server.
- If using TMS Extension for Microsoft Exchange, use its configuration tool to point to the new location of the TMS server.
- If replication is failing due to DNS lookups, make sure that the FQDN configured at the bottom of the TMSAgent settings page of the TMS (typically the FQDN of the TMS as configured in Windows) is resolvable to the ip address of the TMS when the VCS does a lookup on this FQDN. Reverse lookups must also be present.

End of Document