USER SOFTWARE WORKS

TSMPoint

Version 1.0.1

Planning and Installation Guide

CONTENTS

1. Introduction to TSMPoint

After a successful Microsoft® SharePoint® implementation, if the users realize its potential (versioned storage, etc.), the volume of stored documents could rise exponentially. This is the so called ¨SharePoint boom".

Having a very large database can have a negative impact on business continuity and maintenance operations.

- Backup and restore operations take considerably longer
- Index and statistics defragmentation takes considerably longer. This is a particular concern if the database must be taken offline during defragmentation.
- Regular Database Console Commands consistency checks will take much longer. If database integrity is not regularly monitored, the risk of a corrupted database is considerably increased. Larger databases will have a higher risk of corruption due to physical storage errors simply because of the large quantity of storage they consume.

For these reasons, enabling Microsoft Remote Blob Store on an otherwise very large database can be very beneficial as each of the concerns addressed above are alleviated.

Using TSMPoint, only document metadata is stored into the database, documents can be directed into IBM® Spectrum Protect™, eliminating database inflation. Documents get directed to the database or to Spectrum Protect based on their size. Small documents get stored into the database, documents bigger than a configurable size limit get directed to Spectrum Protect.

Benefits

- The decreased size of the SharePoint database allows a higher percentage of it to be kept in memory, increasing its performance.
- The smaller SharePoint database size also speeds up its backup and restore.
- Existing SharePoint databases can be migrated using built-in SharePoint commands.
- Documents stored in Spectrum Protect can be moved back to the Microsoft SQL Server® database with standard methods, should the need for system restructuring arise.

2. Technical Overview of TSMPoint

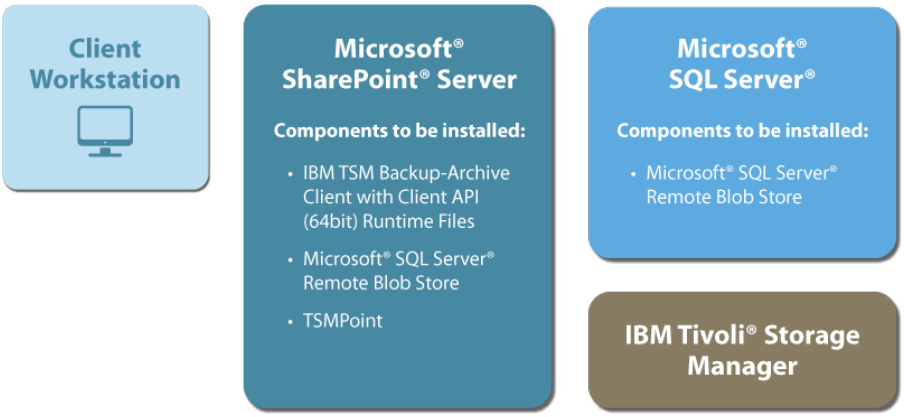
When archiving documents into Microsoft SharePoint, unstructured data or BLOBs are stored into Microsoft SQL Server database. These BLOBs on average take up about 95% of database storage space in Microsoft SQL Server. Using TSMPoint, these BLOBs can be stored in IBM Spectrum Protect (Tivoli® Storage Manager or TSM). TSMPoint uses the Remote Blob Store (RBS) Microsoft SQL Server extension and IBM Spectrum Protect APIs.

2.1 TSMPoint components

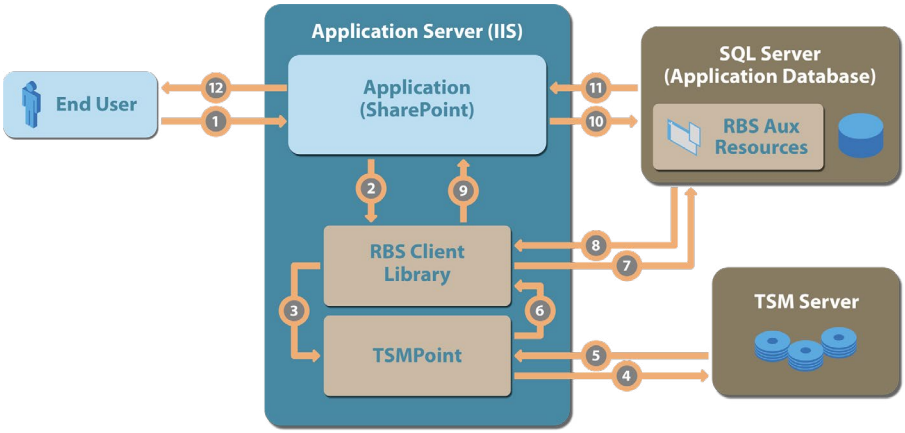
For TSMPoint to work, the following component requirements apply:

- Microsoft SQL Server Blob store component installed on all Microsoft SharePoint Servers
- Microsoft SQL Server Blob Store component installed on all Microsoft SQL Servers
- Spectrum Protect Client API (64 bit) Runtime Files component installed on Microsoft SharePoint Server(s)
- TSMPoint itself installed on Microsoft SharePoint Server(s)

This is shown in the following picture.

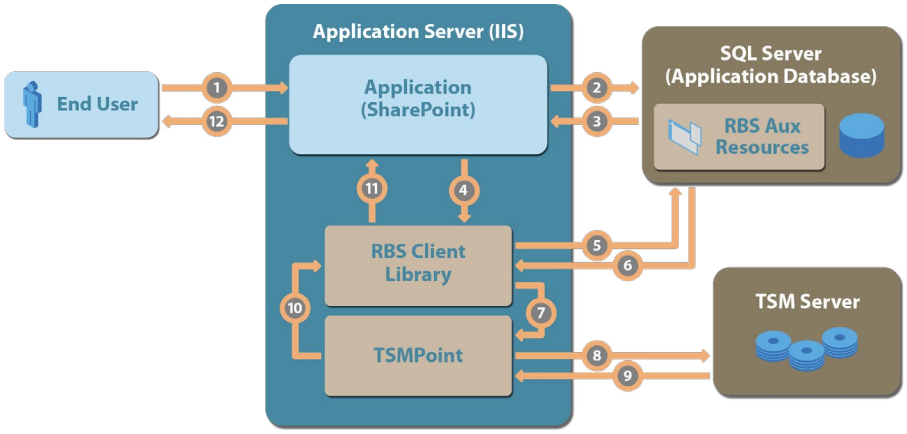


2.2 The BLOB insert process



1. An end user sends a document into the application for processing.
2. The application makes calls to the client library in order to create an RBS session context. For this it uses the connection string to the database containing the RBS auxiliary resources. The application uses the context to call the client library, create a new BLOB object and initiate a BLOB stream write operation.
3. In order to write the BLOB data to the BLOB store the client library invokes the provider library.
4. The provider library writes the BLOB stream to the BLOB store which generates the store_ blob_id. This ID is used in the file name as the BLOB data is stored in the BLOB store.
5. For a CAS device, cloud storage, or other similar BLOB store the BLOB may return a StoreBlobId that is returned to the client library and stored in the RBS auxiliary tables.
6. The client library registers the BLOB in the RBS auxiliary tables in the content database using a BLOB Registration stored procedure.
7. The CollectionId, BLOBStoreId, StorePoolId, StoreBlobId, CreatedTime, and BlobSize are passed into a stored procedure.
8. The registration stored procedure returns the BlobNumber to the client library. The client library constructs a byte array that represents the BlobId using the BlobNumber, CollectionId and additional flags.
9. The BlobId is returned to the application.
10. The application sends the document metadata (including the BlobId) to the database. The BlobId is stored in the registered RBS column and table.
11. The success or failure status of the document insert is returned to the calling application method.
12. The success or failure status of the upload operation to the application is returned to the end user.

2.3 BLOB Retrieve



The process begins with an end user request from the application for a document.

1. The application calls the database for the document's data.
2. The client database returns the document's metadata, including values from the registered RBS column containing the BlobId.
3. Because of the not null value of the registered RBS column, the application knows that it must invoke the RBS client library to retrieve the BLOB data for the document. The application uses the RBS Client Library to create an RBS Session. This session is used to instantiate a BLOB object using the BlobId.
4. During the object instantiation, the RBS Client Library passes the BlobId information to an RBS stored procedure in the client database.
5. The RBS stored procedure returns the information necessary to construct a BlobReference. The StoreBlobId, which is part of the BlobReference, is needed before the BLOB stream can be returned to the application.
6. The application calls for a readable stream through client library.
7. To retrieve the readable stream, the client library invokes the provider library. This library uses the BlobReference in order to construct a request to the BLOB store.
8. The provider library uses te StoreBlobID (from the BlobReference) to send a request for the BLOB stream to the BLOB store.
9. The BLOB store returns the BLOB stream to the provider library.
10. The provider library forwards to the client library the readable BLOB stream.
11. The client library returns the readable BLOB stream to the calling application method.
12. The application streams the document's data to the end user.

3. Planning

3.1. Hardware requirements

TSMPoint does not have additional hardware requirements, it can be installed into any existing Microsoft SharePoint environment.

3.2. Software requirements

- Microsoft SQL Server 2008 R2 SP3 Enterprise, Datacenter or Developer edition (Build version: 10.0.5500.0)
- SharePoint 2010 (Build version: 14.0.4763.1000) or higher. Installation of the most recent cumulative update and service pack is recommended.
- Remote Blob Store Feature (RBS) installed and configured on every SharePoint and Microsoft SQL Server of the TSMPoint enabled environment.
- Tivoli Storage Manager Client API version 6.4.0.0 or higher installed and configured on every SharePoint Server of the TSMPoint enabled environment.

3.3. Supported operating systems

TSMPoint is supported on all operating systems supported by Microsoft SharePoint.

4. Pre-installation Steps

Before installing TSMPoint, certain Tivoli Storage Manager settings must already be configured. Please contact your Tivoli Storage Manager administrator to set up the environment.

4.1. Setting up Tivoli Storage Manager Server

Because all backup objects inserted into the Tivoli Storage Manager backup storage pool have unique file names generated by TSMPoint, they never expire. Tivoli Storage Manager uses management classes for managing archived and backed up data. TSMPoint uses that management class which selected during the TSMPoint installation. It is recommended to create a Policy Domain for RBS data and register all TSMPoint nodes to this Policy Domain. TSMPoint provides a delete function to remove objects no longer needed from the Tivoli Storage Manager server. For this function to work, the VERDELETED and RETONLY Backup Copy Group parameters must be set to 0, and the TSMPoint nodes must be registered with the BACKDELETE=yes parameter. TSMPoint does not use the VEREXISTS or RETEXTRA Backup Copy Group parameters, they can be left on their default values. An Archive Copy Group is not required, it can be left undefined.

The following example can be used as a template for setting up the TSM environment. (Bold parameters are required.):

```
DEFine DOmain TSMPOINT DESCription="TSMPoint domain"
DEFine POlicyset TSMPOINT STANDARD DESCription="TSMPoint editable policyset"
DEFine MGmtclass  TSMPOINT STANDARD TSMPOINT DESCription="TSMPoint default MGMTC"
DEFine COpygroup  TSMPOINT STANDARD TSMPOINT Type=Backup VERExists=1
VERDeleted=0 RETExtra=0 RETOnly=0  DESTination=STORAGEPOOL
ASsign DEFMGmtclass TSMPOINT STANDARD TSMPOINT
ACTivate POlicyset TSMPOINT STANDARD
REGister NODe TSMPOINT P@ssw0rd DOmain=TSMPOINT BACKDELete=yes
```

4.2. Creating a master key for the Microsoft Remote Blob Store feature

Before installing RBS, the content database must already exist and have a master key. This master key can be created using the following SQL command in the Microsoft SQL Server Management Studio.

```
create master key encryption by password = N'<password>'
```

5. Installing and enabling TSMPoint

5.1. Installing Microsoft SQL Server Remote Blob Store feature (RBS.msi)

RBS.msi is part of the Microsoft SQL Server feature pack. You can download the latest version from the official Microsoft SQL Server site (https://www.microsoft.com/en-us/download/details.aspx?id=46696) or use the one included in the Microsoft SQL Server installer package.

It is recommended to install RBS using the silent installation method, because the graphical installation wizard sets certain default values that are not recommended for SharePoint. For more information, refer to Microsoft's recommendations found at https://technet.microsoft.com/en-us/library/ff629463.aspx

The account performing the installation must be a member of the Administrators operating system group.

Note: the installation may take a few minutes.

5.1.1. RBS silent installation on Microsoft SQL Server

In addition to being a member of the Administrators operating system group, the account performing the installation must have the **dbcreator, public** and **securityadmin** Microsoft SQL Server roles.

The RBS silent installation is performed by executing the following command in a Windows® command line prompt:

```
start /wait msiexec /qn /lvx* <log file> /i <RBS.msi>
REMOTEBLOBENABLE=1 DBNAME=<content database> DBINSTANCE=<instance>
ADDLOCAL="Client,ServerScript,EnableRBS"
```

Where

- **<log file>** is the log file containing all installation messages.
- **<RBS.msi>** is the installation package.
- **<content database>** is the name of the content database.
- **<instance>** is the name of the Microsoft SQL Server instance.

For example:

```
start /wait msiexec /qn /lvx* rbs_install_log.txt /i RBS.msi REMOTEBLOBENABLE=1
DBNAME="SP001" DBINSTANCE="SQL" ADDLOCAL="Client,ServerScript,EnableRBS"
```

Note: RBS must be installed for each content database residing on a Microsoft SQL Server.

5.1.2. RBS silent installation on Microsoft SharePoint Server

The RBS silent installation is performed by executing the following command in a Windows command line prompt:

```
start /wait msiexec /qn /lvx* <log file> /i <RBS.msi> DBNAME=<content
database> DBINSTANCE=<instance> ADDLOCAL="Client,Maintainer,ServerScript"
TEXTFILEVALUE="7" CONSOLEVALUE="7" DBTABLEVALUE="7" EVENTLOGVALUE="7"
```

Where

- **<log file>** is the log file containing all installation messages.
- **<RBS.msi>** is the installation package.
- **<content database>** is the name of the content database.
- **<instance>** is the name of the Microsoft SQL Server instance.

For example:

```
start /wait msiexec /qn /lvx* rbs_install_log.txt /i RBS.msi DBNAME="SP001"
DBINSTANCE="SQL" ADDLOCAL="Client,Maintainer,ServerScript" TEXTFILEVALUE="7"
CONSOLEVALUE="7" DBTABLEVALUE="7" EVENTLOGVALUE="7"
```

5.1.3. Checking the RBS installation

After a successful installation:

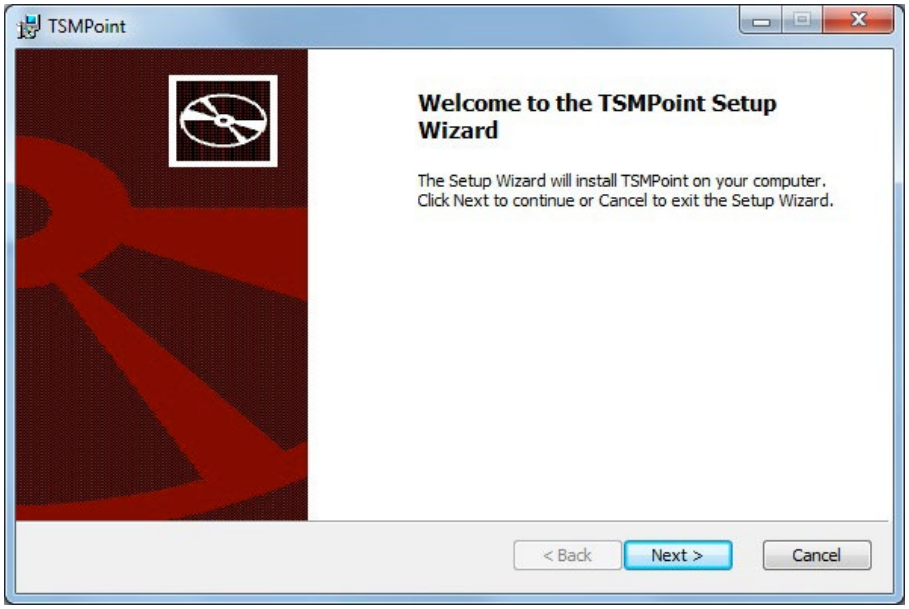
- The last lines of the installation log file contain the following line:
  `Product: Microsoft SQL Server <version number> Remote BLOB Store -- Installation completed successfully.`
- New tables get created in the content database in the `mssqlrbs_resources` schema.
  The following SQL query can be used to check if these tables exist:
  `SELECT [config_key],[config_value] FROM [<content database>].[mssqlrbs_resources].[rbs_internal_config]`
  For example:
  `SELECT [config_key],[config_value] FROM [SP001].[mssqlrbs_resources].[rbs_internal_config]`
- A folder called `Microsoft SQL Remote BLOB Store <version number>` is created in the `Program Files` folder.

If any of the above is missing, the installation was unsuccessful.

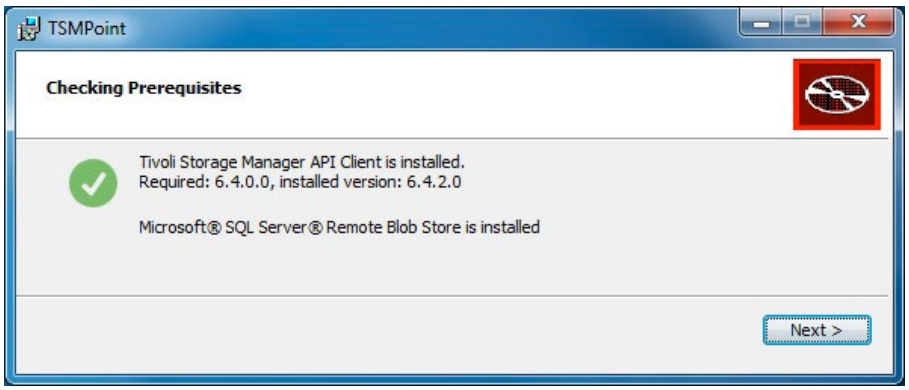
5.2. Installing TSMPoint on Microsoft SharePoint Server(s)

Note: Before installing TSMPoint, TSM Client API (64 bit) Runtime Files component must be already installed.

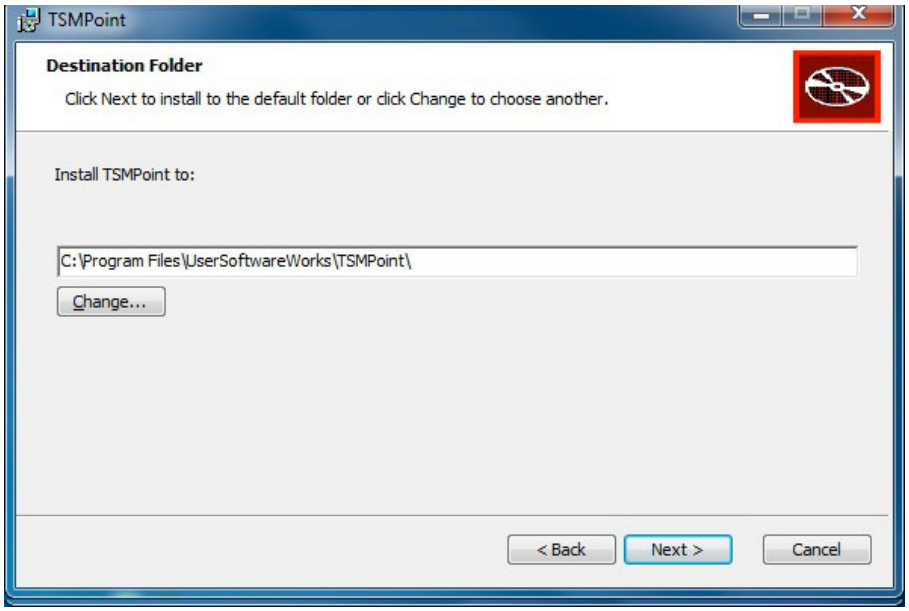
Installing TSMPoint is performed by executing the installer wizard (TSMPoint_Installer.msi), which will guide you through the installation.



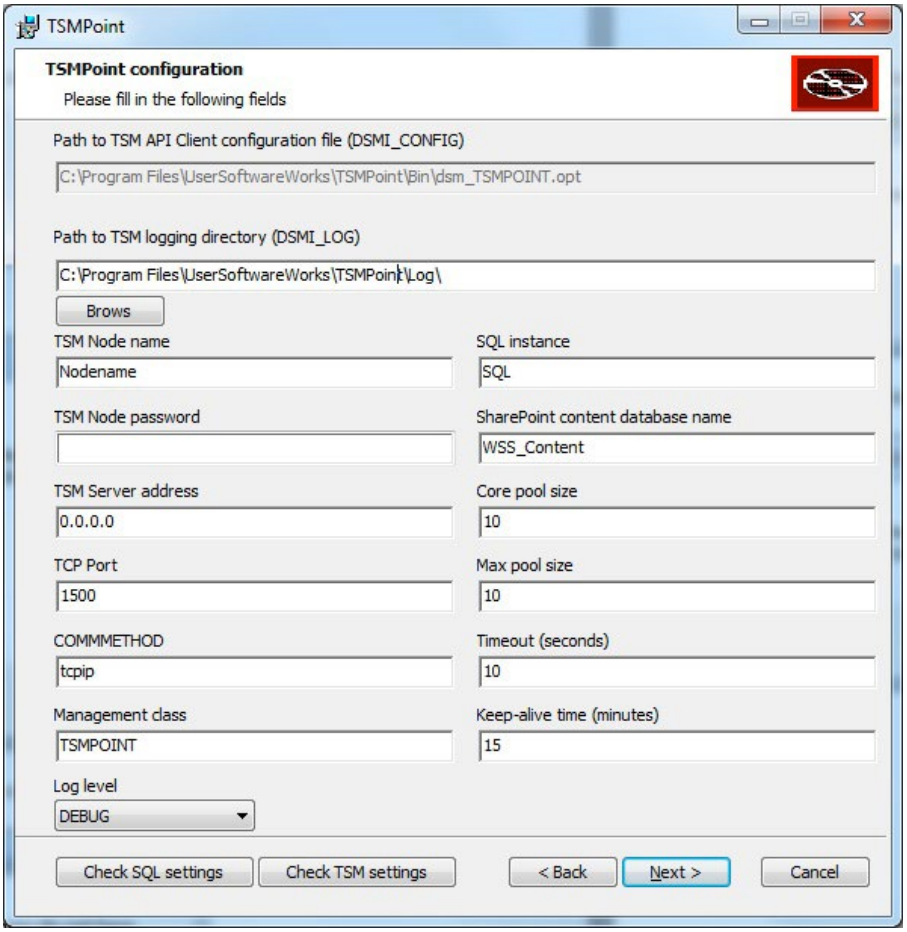
The installer will check for its Tivoli Storage Manager Client and Microsoft SQL Server Remote Blob Store prerequisites.



On the Destination Folder screen, specify the install location for TSMPoint and click Next.



TSMPoint installation configuration.



Parameters to be speficied:

Parameter	Description
Path to TSM logging directory (DSMI_LOG)[1]	The directory to use for logging.
TSM Node name[2]	TSM name of the SharePoint Server as described in chapter 4.1.
TSM Node password[2]	Password of TSM node.
TSM Server address[3]	Specifies the TCP/IP address of a Tivoli Storage Manager server.
TCP Port[3]	Specifies a TCP/IP port of a Tivoli Storage Manager server.
COMMMETHOD[3]	Specifies the communication method you use to provide connectivity for client-server communication.
Management class[3]	Defines a storage policy for the data.
Log level[4]	TSMPoint log level.
SQL Instance	The name of the database server.
SharePoint content database name	The name of the content database.

1. This environment variable can be modified in Windows Control Panel.
2. To modify this parameter, the `extended configuration` field of `[mssqlrbs_resources].[rbs_internal_blob_stores]` must be updated.
3. This parameter can be modified in the **dsm_TSMPoint.opt** file.
4. To modify this parameter, the line `log4cpp.rootCategory=<LOGLEVEL>, A3` of the TSMPoint log configuration file must be updated. This file can be found in the Log subdirectory of TSMPoint installation. The value of `<LOGLEVEL>` can be `INFO`, `ERROR` or `DEBUG`.

Parameter	Description
`Core pool size`[2]	This field specifies the minimum number of TSM Client sessions.
`Max pool size`[2]	This field specifies the maximum number of TSM Client sessions.
`Timeout`[2]	The period TSMPoint waits for a session. If no sessions are available during this period, error is displayed
`Keep-alive time`[2]	These field specifies the period after sessions are reinitialized.

During TSMPoint installation, several Tivoli Storage Manager parameters must be configured. These parameters such as TSM server address, port and communication method for connecting to the Tivoli Storage Manager server are provided by your TSM administrator.

These values, together with other parameters, are stored in a Tivoli Storage Manager options file. By default the options file is named dsm.opt. The DSMI_CONFIG environment variable which has been set during TSMPoint installation refers to this file.

Tivoli Storage Manager uses management classes to manage backups on the Tivoli Storage Manager server. When you store an object with TSMPoint, the default management class for your node is used.

As described in chapter *4.1 Setting up Tivoli Storage Manager Server,* it is recommended to define a separate policy domain where the default management class has the required settings. All TSMPoint nodes should be registered to this policy domain.

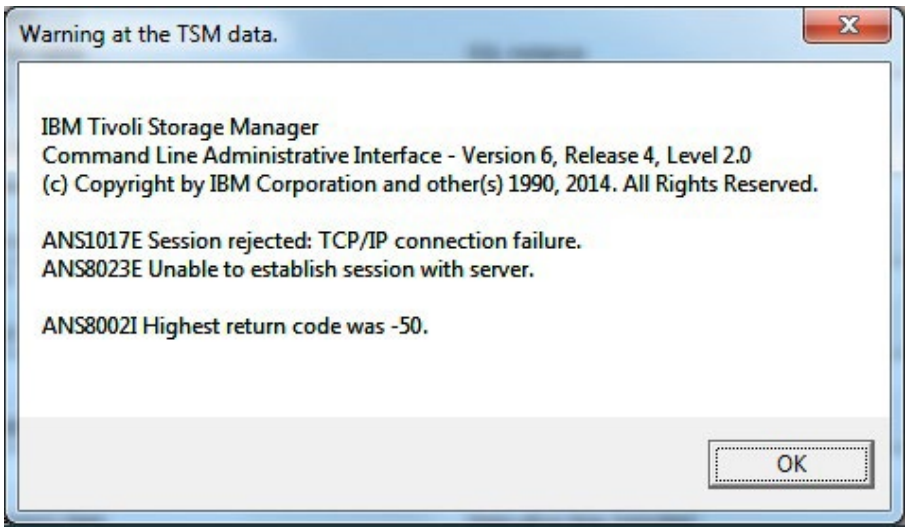
If the management class used is not the default for that domain, an INCLUDE statement must be added to the TSMPoint options file to direct all objects to that management class.

For example:
To assign a management class name `mymanagementclass` to all RBS backups, add this INCLUDE statement to the **dsm.opt** file:

```
include * mymanagementclass
```

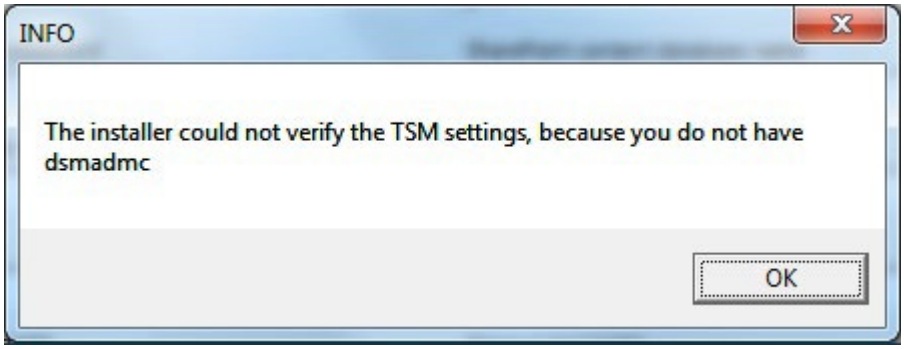
For additional information on creating the **dsm.opt** file and available options, see IBM Tivoli Storage Manager for Windows Backup-Archive Installation and User's Guide.

You can check the TSM server settings (see Chapter 4.1) by clicking the “Check TSM settings” button. If some of the TSM setup parameters are not correct, a warning message box will appear with the corresponding error message(s).



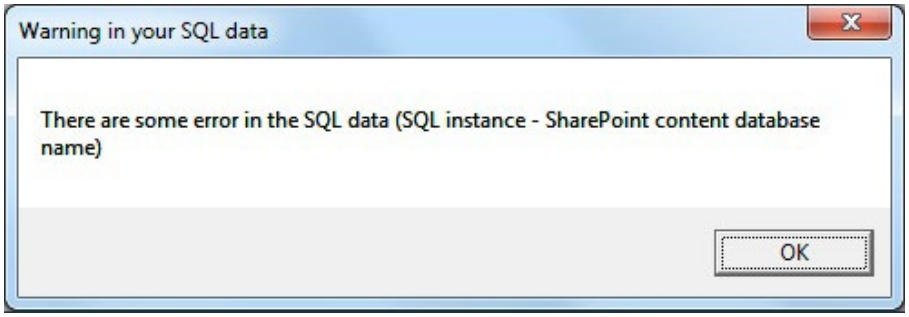
2. To modify this parameter, the `extended configuration` field of `[mssqlrbs_resources].[rbs_internal_blob_stores]` must be updated.

If you don't have dsmadmc, the installer will display an info message.

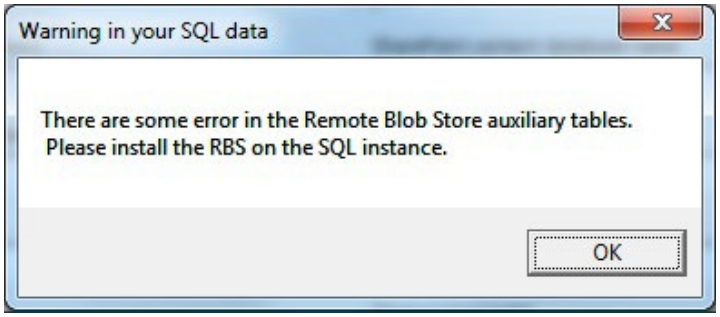


If all of the TSM data are correct, the installer will display an info message.

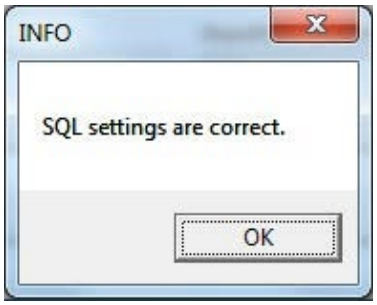
You can also check your content database settings by clicking the "Check SQL settings" button. If one of the `SQL instance` or `SharePoint content database name` paramaters are not valid, a warning message box will appear with the following error message.



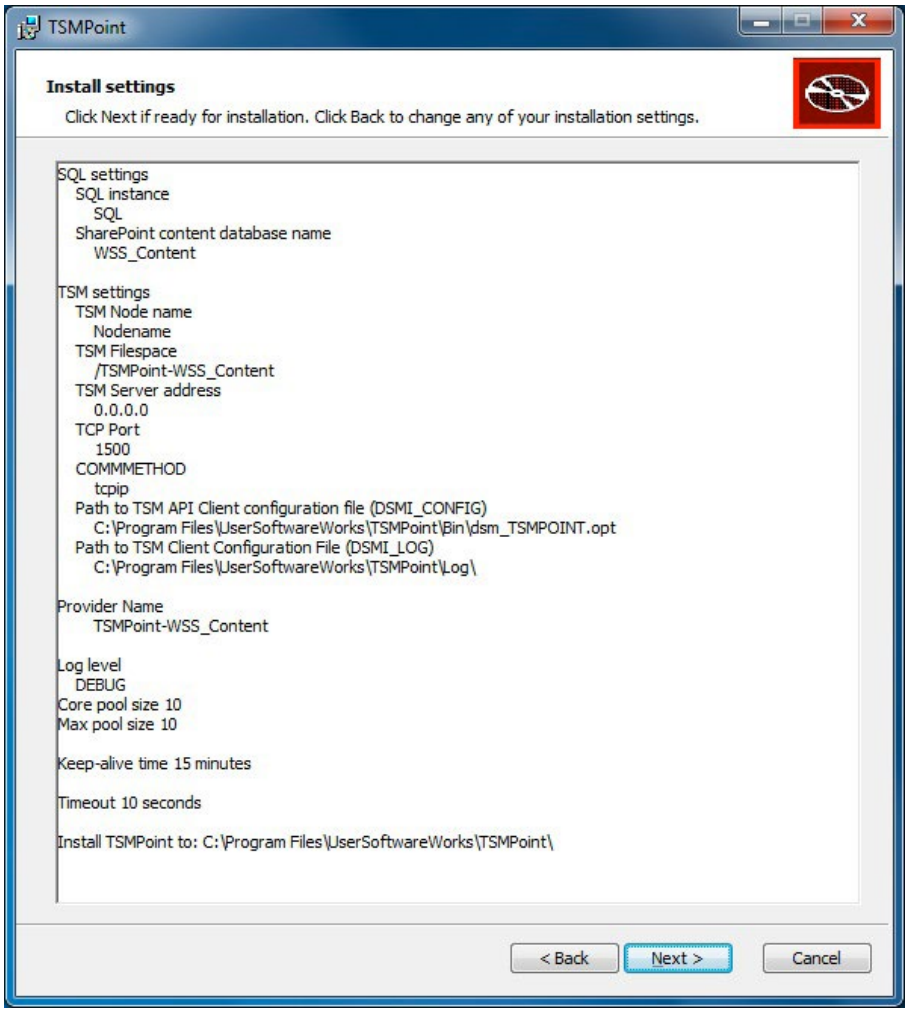
If you didn't install the Remote Blob Storage on the SQL instance, the following warning message will appear.



If all of the SQL data are correct, the installer will display an info message.

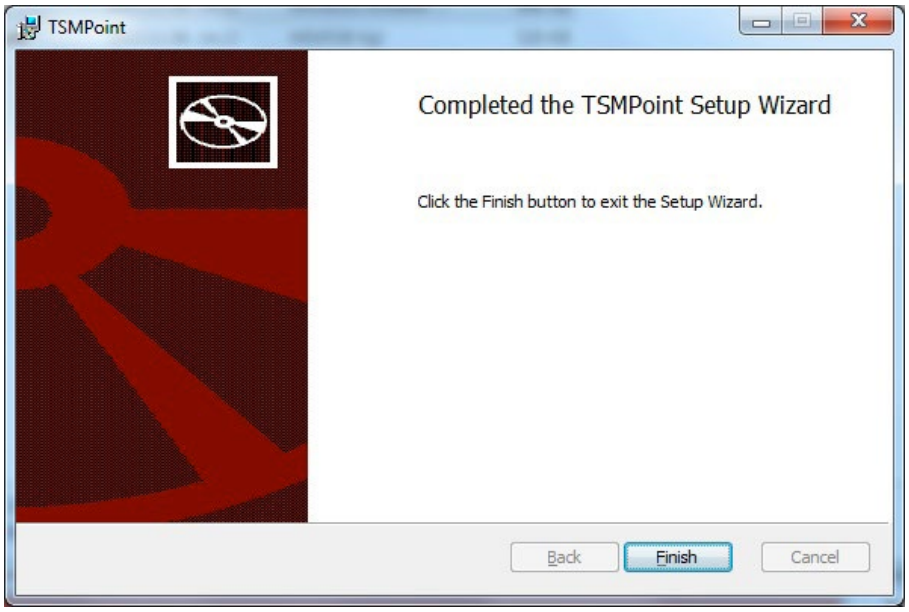


After verifying the settings, click Next to start the installation.



Caution
The name of TSM filespace used by TSMPoint is set by the installer and stored in the database among other configuration parameters. If this parameter is modified manually, RBS Maintainer will delete all non-TSMPoint stored documents from this TSM filespace on its first run. **To avoid problems like this, manual modification of installation parameters is highly not recommended.**

The installation of TSMPoint is completed.



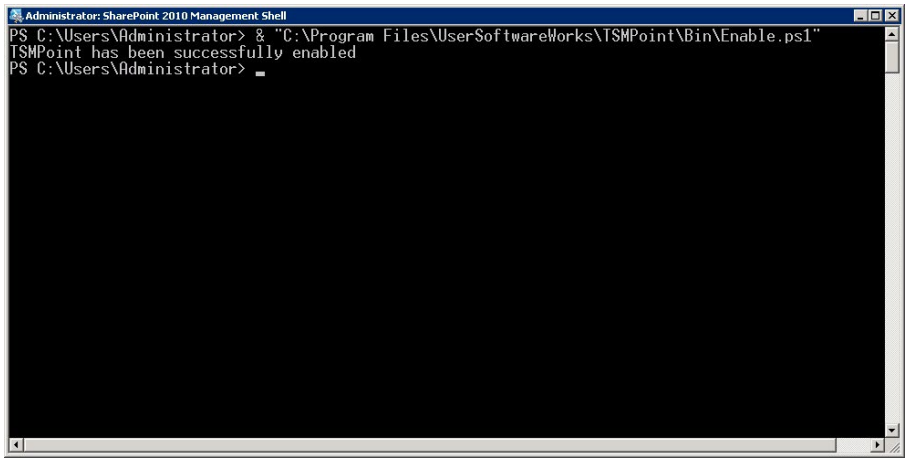
5.3. Enable TSMPoint

Note: this step is mandatory.

To enable TSMPoint, start the SharePoint Management Shell and execute the `Enable.ps1` script found in the bin subdirectory of TSMPoint's installation directory.

For information on starting SharePoint Management Shell, refer to:
https://technet.microsoft.com/en-us/library/ee806878.aspx

For example:



5.3.1. Multiple content databases

If there are multiple content databases, the silent install of Remote BLOB Store must be run for all of them on the SQL Server, using the appropriate database names as its parameter. (See chapter 5.1.1.)

On the Sharepoint server, the InstallProvider.cmd and Enable.ps1 scripts must be run to enable TSMPoint functionality for a content database. Both scripts must be edited before execution.

In InstallProvider.cmd the following lines must be modified:

```
set ProviderName=TSMPoint-databasename
set DatabaseName=databasename
set FSNAME=/TSMPoint-databasename
```

In Enable.ps1 only one line must be modified:

```
$rbss.SetActiveProviderName("TSMPoint-databasename")
```

After editing the scripts, first InstallProvider.cmd then Enable.ps1 must be executed.

The scripts have to be customized and executed for each database.

6. Post-installation Steps

Before using TSMPoint, it is recommended to execute the following configuration steps.

6.1. Checking the TSMPoint installation

Run the `CheckInstallation.exe` in a command line prompt. This application checks TSMPoint functionality by uploading a file. The program have five parameters:

- site name
- user name
- password
- user domain
- upload folder (optional, defaults to Shared Documents)

For example:

```
CheckInstallation.exe http://spw2k08sp/sites/sp001/ Administrator Passw@rd1
SPW2K08SP
```

Note: After Checkinstallation.exe has run successfully, it is vital to verify that a file named TSMPoint.log got created under the Log directory of TSMPoint installation directory. If this file does not exist, the document stored and restored by the Checkinstallation.exe utility was processed without TSMpoint. This phenomenon occurs when Enable.ps1, mentioned in chapter 5.3, has not run.

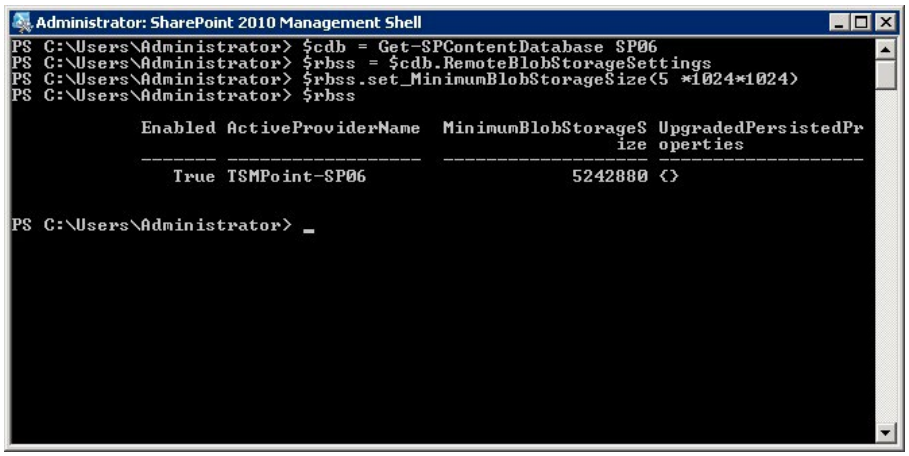
6.2. Setting the Minimum Blob Store Size

It is recommended to alter the default value of Minimum Blob Store Size threshold of RBS according to the environment. BLOBs larger than the threshold will be stored in TSM, other, smaller BLOBs will remain in the content database.

To set and/or check the Minimum Blob Store Size threshold, start the SharePoint Management Shell and execute the following commands:

```
$cdb = Get-SPContentDatabase <content database>
$rbss = $cdb.RemoteBlobStorageSettings
$rbss.set_MinimumBlobStorageSize(<threshold size in bytes>)
$rbss
```

For example:



6.3. Configuring the Microsoft Remote BLOB Store Maintaner

RBS Maintainer performs periodic garbage collection in an RBS deployment. This task must be scheduled using Windows Task Scheduler or, alternatively, any other scheduler that can execute it on a regular basis.

To configure RBS Maintainer, please follow Microsoft's instructions:
https://technet.microsoft.com/en-us/library/gg316773%28v=sql.105%29.aspx

7. Migrating

If the content database already contains BLOBs, they can be migrated to TSM with the rbss.Migrate() command. While migration can be performed at any time, it is recommended to perform it during low utilization periods, so that it doesn´t cause performance problems. Migration moves appropriately sized BLOBs from the specified content database to TSM.

More information:
https://technet.microsoft.com/en-us/library/ff628254.aspx

For the migration, the TSMPoint provider name, which is set by the TSMPoint installer, must be provided. This information can be retrieved from the content database by executing the following SQL query in the Microsoft SQL Server Management Studio:

```
SELECT [blob_store_name] FROM [<content database>].[mssqlrbs_resources]
.[rbs_internal_blob_stores]
```

For example:

```
SELECT [blob_store_name] FROM [SP001].[mssqlrbs_resources]
.[rbs_internal_blob_stores]
```

7.1 Migrating BLOBs to TSM

The following commands will migrate BLOBs to the TSM. The commands must be run in SharePoint Management Shell.

```
$cdb = Get-SPContentDatabase <content database>
$rbss = $cdb.RemoteBlobStorageSettings
$rbss.SetActiveProviderName("<TSMPoint provider name>")
$rbss.Migrate()
```

For example:

```
$cdb = Get-SPContentDatabase SP001
$rbss = $cdb.RemoteBlobStorageSettings
$rbss.SetActiveProviderName("TSMPoint-SP001")
$rbss.Migrate()
```

7.2 Migrating BLOBs back to the SQL database

BLOBs can be migrated back to the SQL database by setting the active provider name to the empty string and running the `rbss.Migrate()` command in SharePoint Management Shell.

```
$rbss.SetActiveProviderName("")
$rbss.Migrate()
```

8. Uninstalling

Every component should be uninstalled via Control Panel – Programs and Features.

Caution
Before uninstalling Microsoft Remote Blob Store, TSMPoint or TSM API Client the Blob store must be empty. In order to achieve this, you must migrate the BLOBs back from the TSM Server to the SQL Database. Failure to do so will result in data loss!

Note: Content migration back to the SQL Database will not delete data from the TSM Server. It is done by the RBS Maintainer which runs periodically. If you would like to force the run of Microsoft RBS Maintainer, please refer to Microsoft's documentation:
https://technet.microsoft.com/en-us/library/gg316773%28v=sql.105%29.aspx

9. Troubleshooting

9.1 Installation

9.1.1 RBS

If the installation was unsuccessful, contact your Microsoft SQL Server and Microsoft SharePoint Support, providing them the contents of the installation log file.

9.1.2 TSMPoint

If TSMPoint installation was unsuccessful, check the installation log file found in the %TEMP% folder. The installer does not proceed until all of TSMPoint prerequisites, namely Tivoli Storage Manager Client and Microsoft SQL Server Remote Blob Store, are installed. Please install the missing prerequisite(s) before installing TSMPoint.

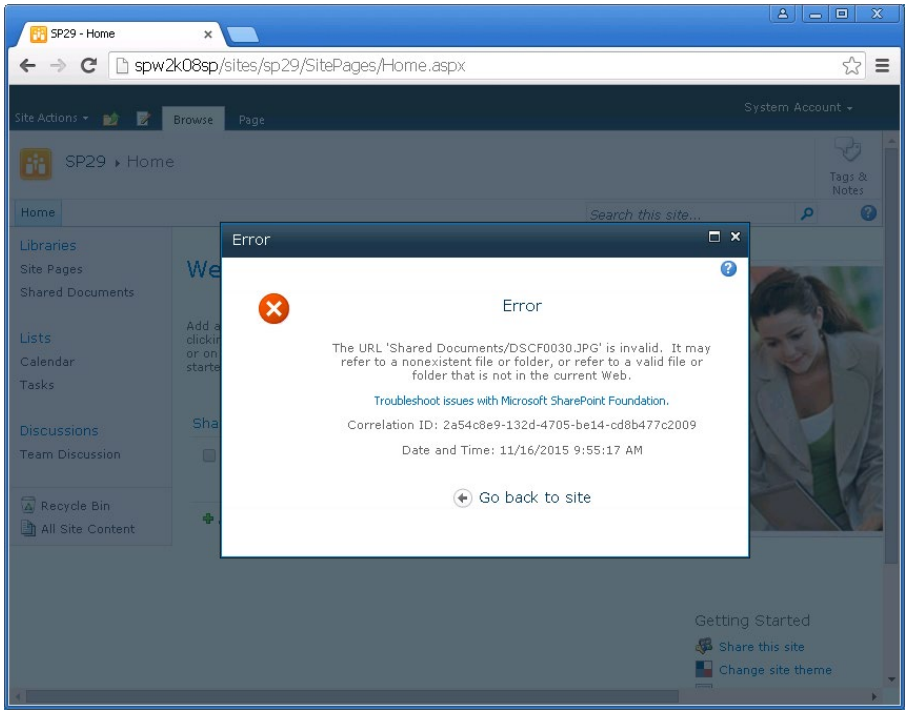
TSMPoint installer will, at the end of the installation, execute InstallProvider.cmd, which registers TSMPoint into the given content database and into the configuration file of Microsoft SharePoint.

9.1.3 TSM API Client

If the TSM Client API installation was unsuccessful, contact your TSM administrator or refer to IBM Tivoli Storage Manager for Windows Backup-Archive Clients – Installation and User's Guide.

9.2 Runtime

In case of a runtime error (for example a document upload or a download) the SharePoint application server shows an error dialog in the browser:



9.2.1 SharePoint application server logs

The primary troubleshooting option is the SharePoint application server logs.

Depending on the version of SharePoint, the logs can be found in the directory:

- SharePoint 2010
  - C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\LOGS
- SharePoint 2013
  - C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\15\LOGS

In case of an unsuccessful TSMPoint.dll assembly load, the following entry is inserted into the SharePoint application server log:

Exception thrown storing stream in new SqlRemoteBlob: System.Reflection.TargetInvocationException: Exception has been thrown by the target of an invocation. ---> Microsoft.Data.SqlRemoteBlobs. RemoteBlobConfigurationException: No provider of type <User TSMPoint> found. Check the server configuration or install the provider on the client.

If the error is not RBS specific, please contact your SharePoint application server administrator or Microsoft Support, providing them the contents of SharePoint application server log.

All messages and errors are logged by default

There is a very robust event throttling section in Central Administration that enables customizing the logs to the area of the issue is in and, once the problem is solved, dialing it back easily.

In Central Administration, click Monitoring, then select Configure Diagnostic Logging under the Reporting section.

More information:
https://msdn.microsoft.com/en-us/library/office/gg193966(v=office.14)

9.2.2 Microsoft Remote Blob Store logs

Further details can be acquired from the Microsoft Remote Blob Store log files under the SharePoint application server home directory:

- `C:\inetpub\wwwroot\wss\VirtualDirectories\80\log\RBSLog_<DateTime>.txt`

In case of an unsuccessful TSMPoint.dll assembly load, the following entry is inserted into the Microsoft RBS log:

```
4 :09:17:52.877:13:ERR :0 : : :AssemblyLoad  : An error occurred while reading
extension file C:\Program Files\UserSoftwareWorks\TSMPoint\Bin\TSMPoint.dll.
4 :09:17:52.877:13:ERR :0 : : :AssemblyLoad  : System.IO.FileNotFoundException:
```

The system cannot find the file specified. (Exception from HRESULT: 0x80070002)

9.2.3 TSMPoint logs

TSMPoint log location and log level is specified in the TSMPointLOG.properties file.

This file can be found in the Log subdirectory of TSMPoint's installation folder.

Its default location is:

- `C:\Program Files\UserSoftwareWorks\TSMPoint\Log\TSMPointLOG.properties`

The most frequent TSMPoint exceptions and errors are:

Error message	Description
All sessions are in use	All sessions are in use. When a session is released, this error automatically resolves. If this message occurs frequently, increase the Core pool size és Max pool size parameter values in the configuration table of the content database ([mssqlrbs_resources].[rbs_internal_blob_stores]), then restart the application (IISRESET).
Extended configuration parameter is null	The extended configuration field is empty in the [mssqlrbs_resources] .[rbs_internal_blob_stores] table.
File Space Name is null	The fileSpaceName field is missing from the [mssqlrbs_resources] .[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)
Node Name is null	The nodeName configuration item is missing from the [mssqlrbs_ resources].[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)

Error message	Description
Node Password is null	The nodePassword configuration item is missing from the [mssqlrbs_resources].[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)
Core pool size is less than 1	The value of the corePoolSize configuration item is less than one in the [mssqlrbs_resources].[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)
Max pool size is less than Core pool size	The value of the maxPoolSize configuration item is less than the value of the corePoolSize configuration item in the [mssqlrbs_resources].[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)
Keep-alive time is negative	The value of the keepAliveTime configuration item is negative in the [mssqlrbs_resources].[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)
Timeout is negative	The value of the timeout configuration item is negative in the [mssqlrbs_resources].[rbs_internal_blob_stores] table. The extended configuration field must be modified and the application restarted. (IISRESET)
Parameter received from SharePoint is null	One or more parameters received from SharePoint are null. The application must be restarted. (IISRESET) If the error persists, contact the SharePoint administrator.
Log configuration error. Missing TSMPoint_LOG environment variable or log configuration file	Either the TSMPoint_LOG environment variable or the TSMPointLOG.properties file is missing. Set the TSMPoint_LOG environment variable to the directory where the TSMPointLOG.properties file resides. This message gets inserted into the SharePoint log.
Application version is not appropriate	The TSM API Client version is not supported. It must be upgraded to a supported version.

If a log entry has an RC code, it refers to the TSM API Client error log. For further information, consult the TSM API Client error log section.

For example, in case of an unsuccessful connection to the TSM Server, the following entry is found in the TSMPoint.log file:

```
[ERROR] InitializeSession failed. RC:-50
```

9.2.4 TSM API Client error logs

During TSMPoint installation the DSMI_LOG variable must be specified. The TSM API Client error logs can be found there. The name of the log file is dsierror.log

An example error message in dsierror.log:

ANS5216E Could not establish a TCP/IP connection with address '0.0.0.0:1500'. The TCP/IP error is 'A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond.' (errno = 10060).
ANS9020E Could not establish a session with a TSM server or client agent. The TSM return code is -50.

To enable tracing for the TSM API, the following lines have to be added to the dsm.opt file or the file designated as the client options file:

```
TRACEFILE trace file name TRACEFLAGS trace flags
```

Where:

- **`trace file name`** – The name of the file where the trace data is written.
- **`trace flags`** – The list of trace flags to enable,separated by a space.

The trace flags specific to the TSM API:

- **api** – Basic information about the API function calls
- **api_detail** – Detailed information about the API function calls

Other TSM Backup/Archive client and TSM API trace flags can also be specified. For a list of all trace classes available, refer to Backup/Archive client.

For example:

```
TRACEFILE C:\Program Files\UserSoftwareWorks\TSMPoint\Log\trace.out TRACEFLAGS
api api_detail verbinfo verbdetail timestamp
```

More information:
https://publib.boulder.ibm.com/tividd/td/TSMM/SC32-9103-01/en_US/HTML/cli_api.html

In case of a TSM API Client error, please contact your TSM Server administrator.

9.2.5 Microsoft SQL Server logs

Trace files can be generated using SQL Server Profiler, found in the Program Files (x86)\Microsoft SQL Server\100\Tools\Binn subdirectory of the Microsoft SQL Server installation. Trace files are generated by executing the following:

1. On the File menu, click New Trace and connect to an SQL Server. The Trace Properties dialog box appears. Note: If Start tracing is selected immediately after making connection, the Trace Properties dialog box fails to appear and the trace begins instead. To turn off this setting, click Options on the Tools menu and clear the Start tracing immediately after making connection check box.
2. Type a name for the trace in the Trace name box.
3. In the Use the template list, select a trace template on which to base the trace or, if a template is not needed, select Blank.
4. To save the trace results, do one of the following:
   a. Click Save to file to capture the trace to a file. Specify a value for Set maximum file size. The default value is 5 megabytes (MB).
   b. Optionally, select Enable file rollover to automatically create new files when the maximum file size is reached. Server processes trace data can optionally be selected, which causes the service that is running the trace to process trace data instead of the client application. When the server processes trace data, no events are skipped even under stress conditions, but server performance may be affected.
   c. Click Save to table to capture the trace to a database table.
   d. Optionally, click Set maximum rows and specify a value.

Caution
When the trace results are not saved to a file or table, you can view the trace only while SQL Server Profiler is open. However, the trace results are lost after the trace is stopped and SQL Server Profiler is closed. To avoid losing the trace results, click Save on the File menu to save the results before closing SQL Server Profiler.

5. Optionally, select the Enable trace stop time check box, and specify a stop date and time.
6. To add or remove events, data columns or filters, click the Events Selection tab. For more information, see: How to: Specify Events for a Trace File (SQL Server Profiler)
7. Click Run to start the trace.

More information:
https://msdn.microsoft.com/en-us/library/ms175047(v=sql.105).aspx

9.2.6 How to get TSMPoint support

If a TSMPoint error occurs

- Turn on component traces as described before
- Reproduce the error
- Collect all log files (TSMPoint log file, SharePoint log file, TSM API Client trace file,TSM API error file (dsierror.log), MS SQL trace file) and pack them into a file named <organization>_<yyyymmdd>_<HHmm>.zip.
- E-mail the zip file, information about the environment (such as versions and fixpacks of operating systems, SharePoint Server, Microsoft SQL Server, TSM Server, TSM Client, etc) and a brief description of the situation that could help reproducing the error to **tsmpoint_support@usersoftwareworks.com**

10. Trademarks