



Using Backup Server with
IBM[®] Tivoli[®] Storage Manager

Adaptive Server[®] Enterprise

15.7

DOCUMENT ID: DC01176-01-1570-01

LAST REVISED: September 2011

Copyright © 2011 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

To order additional documents, U.S. and Canadian customers should call Customer Fulfillment at (800) 685-8225, fax (617) 229-9845.

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor. Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase trademarks can be viewed at the Sybase trademarks page at <http://www.sybase.com/detail?id=1011207>. Sybase and the marks listed are trademarks of Sybase, Inc. ® indicates registration in the United States of America.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

IBM and Tivoli are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

CHAPTER 1	Creating Backups Using the IBM Tivoli Storage Manager	1
	Installing and setting up the backup system	2
	Installing the backup system	3
	TSM concepts and Backup Server.....	6
	Logical structures on TSM.....	6
	Object naming and data organization.....	6
	Backing up databases and transactions	7
	Using dump and load with the same and different databases...	8
	Using dump and load when the source and target Adaptive Servers are different	9
	Using dump and load with multiple stripes	10
	Listing a server's backup objects	13
	Deleting backup objects from TSM	13
Index		15



Creating Backups Using the IBM Tivoli Storage Manager

This manual describes how to set up and use the IBM® Tivoli® Storage Manager to backup Adaptive Server® Enterprise databases. Use this manual in conjunction with the Adaptive Server *System Administration Guide* and the relevant Tivoli Storage Manager documentation.

Adaptive Server support for the Tivoli Storage Manager is a licensed option.

Topic	Page
Installing and setting up the backup system	2
TSM concepts and Backup Server	6
Backing up databases and transactions	7
Listing a server's backup objects	13
Deleting backup objects from TSM	13

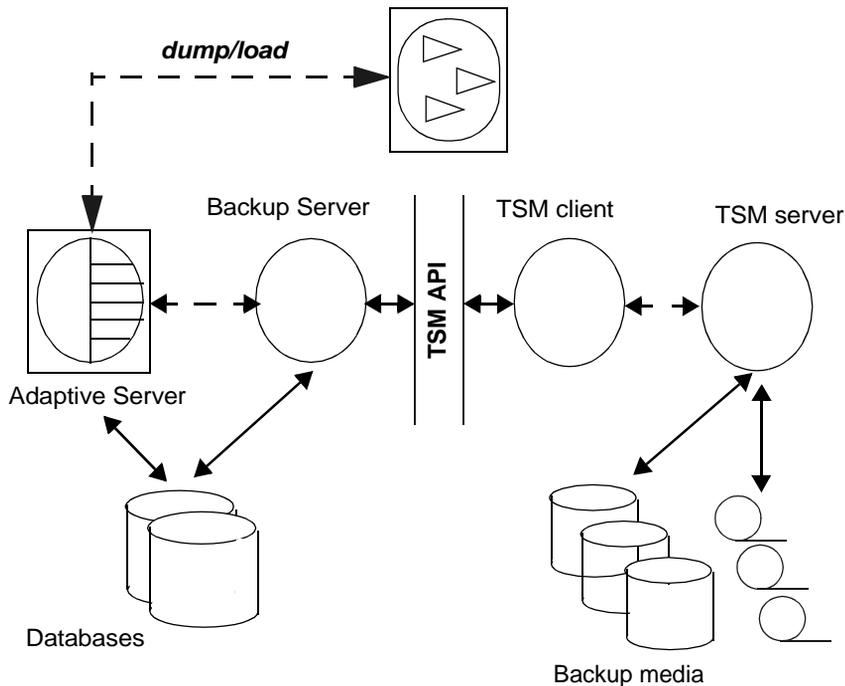
The Tivoli Storage Manager (TSM) is a third-party client/server program that provides storage management services to the licensed user. The Adaptive Server Backup Server supports tape drives and disk files as backup media. TSM works with Backup Server to enable a greater variety of backup media. You can store Adaptive Server backups on any backup media supported by TSM.

Adaptive Server handles the dump and load of databases and transactions to and from TSM; TSM handles storage and retrieval from the storage media.

TSM provides storage management services, but you can administer all your Adaptive Server backup and restore operations, including queries for backup objects and deletion of backup objects, from Backup Server.

Figure 1-1 shows how Adaptive Server works with TSM to back up databases and transactions.

Figure 1-1: Integration of Adaptive Server, Backup Server, and TSM



The Adaptive Server and TSM integrated architecture includes Adaptive Server, Backup Server, the TSM client, the TSM server, and the TSM client API. The TSM client API enables the client application to call TSM functions.

The Sybase® interface with the TSM API provides I/O routines that allow Backup Server to communicate with TSM. When dump or load commands are executed, Backup Server dynamically loads this API module, which in turn loads a shared library supplied by the TSM client API component.

Installing and setting up the backup system

This section describes how to install and configure Backup Server and TSM so they can work together to back up databases and transactions.

Install the TSM client, TSM client API, and the TSM server according to instructions in the TSM documentation. Adaptive Server, Backup Server, and the TSM client API product must be installed on the same machine, and they should be of the same bit architecture as that of Backup Server; the TSM server can be installed on a different machine.

Installing the backup system

After installing Adaptive Server and Backup Server:

- 1 Install and configure the TSM client API on the same machine as Adaptive Server and Backup Server. Install and configure the TSM server on the same or different machine. See the TSM documentation for instructions.
- 2 Set these TSM-specific environment variables on the Backup Server machine:
 - `DSMI_DIR` – TSM installation directory location
 - `DSMI_LOG` – TSM error log location
 - `DSMI_CONFIG` – TSM configuration file
 - `DSMI_LIB` – TSM shared library fully qualified path name

For example, assuming that Adaptive Server, Backup Server, and the TSM client API are installed on a Linux AMD 64-bit machine

- `DSMI_DIR` – `/opt/tivoli/tsm/client/api/bin64`
 - `DSMI_LOG` – `/opt/tivoli/tsm/client/error_log`
 - `DSMI_CONFIG` – `/opt/tivoli/tsm/client/api/bin64/dsm.opt`
 - `DSMI_LIB` – `/opt/tivoli/tsm/client/api/bin64/libAPITSM64.so`
- 3 Update the TSM system file (`dsm.sys`) and the TSM configuration file (`dsm.opt`). At a minimum, set these parameters:
 - `SERVERNAME` – TSM server name.
 - `COMMETHOD` – TSM server communication type.
 - `TCPPORT` – TSM server port number.
 - `TCPSEVERADDRESS` – fully qualified name of TSM server.
 - `NODENAME` – registered name of TSM client.

- `PASSWORDACCESS` – access method type of TSM password. Set to “generate”.
- `PASSWORDDIR` – directory location of TSM password.

Note To improve performance, set the appropriate TSM Client configuration parameters. See the TSM documentation for instructions.

- 4 Create the TSM client node at the TSM server. See the TSM documentation for instructions.
- 5 Use the `sybtsmpasswd` utility to:
 - Register the TSM client node with TSM server.
 - Generate the TSM-encrypted password file on the TSM client node.

`sybtsmpasswd` generates the TSM password file (*TSM.PWD*) in the directory specified by the `PASSWORDDIR` parameter in the TSM configuration file. (See step 3.)

Execute `sybtsmpasswd` while logged in as the operating system user “root.” `sybtsmpasswd` prompts for the existing password and the new password for the registered TSM client node.

Note Execute `sybtsmpasswd` only when you set up TSM or change the TSM client node name, user name, or password.

- 6 Start Backup Server.
- 7 From the TSM server, grant the TSM client node name permission to delete backup copies at the TSM server. See the TSM documentation for instructions.

This enables these Sybase operations at the TSM client node:

- `sp_deletesmobj`
- dump database *database_name* to “`syb_tsm::object_name`” with `init`

After completing these steps, you can execute these commands and stored procedures:

- dump database
- dump transaction
- load database

- load transaction
- sp_querysmobj
- sp_deletesmobj

Configuring TSM to allow different source and target machines

TSM does not automatically allow cross-client or cross-owner backup and restore operations.

For example, suppose you want to dump from machine “node1” and load to machine “node2”. The source client NODENAME is “node1”; the target client NODENAME is “node2”.

To configure TSM to allow this:

- 1 On the TSM client node, set the ASNODENAME parameter to the same value for both source and target client node machines. For example:

```
ASNODENAME MyCluster
```

- 2 On the TSM server node:
 - a Add the client node name. For example: “MyCluster”.
 - b Register the common client node name “MyCluster” association with the source “node1” and target “node2” client nodes. See the TSM documentation for instructions on how to grant proxy authority to a client node.
 - c Grant permission to delete backup copies from the “MyCluster” client node. See the TSM documentation for instruction on how to update TSM client node properties.

Configuring TSM data compression

TSM provides a configuration parameter that enables backup data compression. If compression is enabled, the TSM client compresses the backup data before sending it to the TSM server. This decreases the amount of backup data sent to TSM server and the storage space it occupies at the TSM server.

To enable TSM compression for Sybase database backups, set the “COMPRESSION” parameter to “YES” in the TSM client configuration file (*dsm.sys*). The default value is “NO,” indicating no compression.

See the IBM Tivoli Storage Manager documentation.

TSM concepts and Backup Server

This section describes some key TSM concepts.

Logical structures on TSM

TSM abstracts the details of storage devices into a logical construct that requires few API calls. It allows client applications like Backup Server to use TSM API calls.

TSM uses these hierarchical constructs to structure backup data. Typically, you can use the default values supplied by TSM. Default values can be changed only by the TSM administrator at the TSM server; they cannot be changed from Backup Server.

In descending order, the constructs are:

- Policy domain – the base element of the logical structure, the policy domain associates a TSM client (a node) with a policy set.
- Policy set – each policy domain has one active policy set that determines how TSM server handles data from the TSM clients belonging to the policy domain. Each policy set contains a default management class and one or more additional management classes.
- Management class – backup copy groups that contain information about expiration or retention policies for backup objects.
- Copy group – defines the expiration or retention policies, such as event-based retention, for backup objects.

Note If Adaptive Server backups require special expiration policies, or do not use the default copy groups, the TSM administrator can create custom classes or groups on the TSM server. You cannot create these special management classes and copy groups from Backup Server.

Object naming and data organization

The TSM database schema is optimized for efficient storage and retrieval of data objects. Adaptive Server database or transaction data is backed up as named objects on the TSM server. Backup Server assigns the object names.

Data is organized hierarchically on the TSM server at these levels:

- Filespace
- High-level name
- Low-level name

Backup Server prints the dump's backup object names when you successfully execute `dump database` or `dump transaction`. See “Using dump and load with the same and different databases” on page 8.

Backing up databases and transactions

When you execute the `dump` or `load` command, Backup Server invokes the Sybase interface with TSM API, which provides communication with TSM. When you use the `dump` command, you specify an object name that is uniquely associated with the backup object. This object name is the same as the TSM object name, and should later be used to specify the same database or transaction dump when you execute the `load` command.

In general, you can use the same options with the `dump` and `load` commands with TSM as you use with Backup Server when TSM is not configured.

The `dump` and `load` syntax specific to TSM is:

```
dump {database | transaction} database_name to
"syb_tsm::object_name"

load {database | transaction} database_name from
"syb_tsm::[-S source_server_name]
[-D source_database_name::]object_name"
```

See the *Reference Manual: Commands* for complete syntax and usage information.

The keyword “`syb_tsm`” invokes the Sybase interface with the TSM API (`libsyb_tsm`). When loading a database or transaction, specify a server or database name only when the current server or database is not the same as the server or database associated with the backup object.

Using dump and load with the same and different databases

To dump and then load a database and transaction to a different database, when both databases are on the same Adaptive Server, use the -D option with the load command.

For example, to dump a database “testdb” to a TSM backup object named “obj1.1”, first dump “obj1.1” to the same database, then load “obj1.1” to a different database,

```
dump database testdb to "syb_tsm::obj1.1"
```

Backup Server prints the following, which provides the backup object identifiers:

```
Backup Server session id is: 5. Use this value when
executing the 'sp_volchanged' system stored procedure
after fulfilling any volume change request from the
Backup Server.
```

```
Backup Server: 4.132.1.1: Attempting to open byte stream
device:'syb_tsm::obj1.1::00'
```

```
Backup Server: 6.28.1.1: Dumpfile name
'testdb091840CA13 ' section number 1 mounted on byte
stream 'syb_tsm::obj1.1::00'
```

```
Backup Server: 4.188.1.1: Database testdb: 854
kilobytes (84%) DUMPED.
```

```
Backup Server: 3.43.1.1: Dump phase number 1 completed.
```

```
Backup Server: 3.43.1.1: Dump phase number 2 completed.
```

```
Backup Server: 3.43.1.1: Dump phase number 3 completed.
```

```
Backup Server: 4.188.1.1: Database testdb: 870
kilobytes (100%) DUMPED.
```

```
Backup Server: 3.42.1.1: DUMP is complete (database
testdb).
```

Additional message in Backup Server error log indicating corresponding TSM backup object name and ids. Please check the complete Backup Server error log attached as well.

Check the Backup Server error log to see the TSM backup object name and id:

```
Jul  3 14:30:19 2009: A00: Database 'testdb' dumped.
Tivoli Storage Manager backup object name: fs =
/demo_srv1, high = /testdb.DB, low = /obj1.1.0,
Copyids: (0, 36913).
```

Backup Server also prints the backup object identifiers to the error log. In this example, they are:

- Filespace (fs) – demo_srv1.
- High-level name (high) – testdb.DB.
- Low-level name (low) – obj1.1.0.
- TSM copyid (copyids) – 0, 36913. This is a unique identifier assigned by the TSM server. It has two parts: a high copyid and a low copyid.

To load the backup object “obj1.1.0” to the same database, “testdb,” enter:

```
load database testdb from "syb_tsm::obj1.1"
```

Alternatively, to load the backup object “obj1.1” to another database, “anotherdb,” enter:

```
load database anotherdb from  
"syb_tsm::-D testdb::obj1.1"
```

Using dump and load when the source and target Adaptive Servers are different

To dump and then load the database or transaction to a different database on a different Adaptive Server, use both the -S and -D options with the load command. Make sure that you:

- Configure the TSM server and TSM client to enable the load command for different source and target machines. See “Configuring TSM to allow different source and target machines” on page 5.
- Start Backup Server using the same login on both the source and target machines.
- Use the -S and -D options with the load database or load transaction command, where -S specifies the source server and -D specifies the source database.

```
load database database_name from "syb_tsm::  
-S server_name -D database_name::object_name"
```

For example, to load the backup object “obj1.3” to “anotherdb” where the source server is “myserver” and the database is “testdb,” enter:

```
load database anotherdb from  
"syb_tsm::-S myserver -D testdb::obj1.3"
```

Using dump and load with multiple stripes

When multiple stripes are used with dump database or dump transaction, each stripe creates a separate backup object at the TSM.

You can use the same or different backup object names for multiple stripes when TSM handles the backup media. In this example, each stripe has the same name. TSM uses a low-level identifier to distinguish between each backup object, for example, obj.2. See the error log later in this section.

Note If TSM is supported at your site, you can use the same or different names for each stripe. If TSM is not supported at your site, you must use different names for each stripe.

Using the same object name for each stripe

For example, to dump “testdb” using multiple stripes with the same stripe object name “obj”, enter:

```
dump database testdb to "syb_tsm::obj"  
stripe on "syb_tsm::obj"  
stripe on "syb_tsm::obj"  
stripe on "syb_tsm::obj"  
stripe on "syb_tsm::obj"
```

In this example, Backup Server provides uniqueness for TSM backup objects by appending the stripe identifier to the supplied backup object name “obj”. Check the Backup Server error log to verify that the TSM low-level backup object name is unique for each stripe.

Backup Server prints:

```
Backup Server session id is: 5. Use this value when executing the  
'sp_volchanged' system stored procedure after fulfilling any volume change  
request from the Backup Server.  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj::00'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj::01'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj::02'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj::03'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj::04'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740CDAD ' section number 1  
mounted on byte stream 'syb_tsm::obj::03'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740CDAD ' section number 1
```

```

mounted on byte stream 'syb_tsm::obj::00'
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740CDAD ' section number 1
mounted on byte stream 'syb_tsm::obj::01'
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740CDAD ' section number 1
mounted on byte stream 'syb_tsm::obj::02'
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740CDAD ' section number 1
mounted on byte stream 'syb_tsm::obj::04'
Backup Server: 4.188.1.1: Database testdb: 24 kilobytes (87%) DUMPED.
Backup Server: 4.188.1.1: Database testdb: 106 kilobytes (88%) DUMPED.
Backup Server: 4.188.1.1: Database testdb: 646 kilobytes (97%) DUMPED.
Backup Server: 4.188.1.1: Database testdb: 760 kilobytes (99%) DUMPED.
Backup Server: 4.188.1.1: Database testdb: 874 kilobytes (100%) DUMPED.
Backup Server: 3.43.1.1: Dump phase number 1 completed.
Backup Server: 3.43.1.1: Dump phase number 2 completed.
Backup Server: 3.43.1.1: Dump phase number 3 completed.
Backup Server: 4.188.1.1: Database testdb: 898 kilobytes (100%) DUMPED.
Backup Server: 3.42.1.1: DUMP is complete (database testdb).

```

Check the Backup Server error log to see the TSM backup object names and IDs. Check the complete Backup Server error log attached as well. The Backup Server displays this information:

```

Oct 1 14:37:43 2009: A00: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj.0,
Copyids: (0, 80898).
Oct 1 14:37:43 2009: A03: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj.3,
Copyids: (0, 80897).
Oct 1 14:37:43 2009: A02: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj.2,
Copyids: (0, 80899).
Oct 1 14:37:43 2009: A01: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj.1,
Copyids: (0, 80900).

```

To load “obj” to the same database, enter:

```

load database testdb from "syb_tsm::obj"
stripe on "syb_tsm::obj"
stripe on "syb_tsm::obj"
stripe on "syb_tsm::obj"
stripe on "syb_tsm::obj"

```

Using different object names for each stripe

To dump “testdb” using multiple stripes with different strip object names, enter:

```

dump database testdb to "syb_tsm::obj0"
stripe on "syb_tsm_obj1"

```

```
stripe on "syb_tsm_obj2"  
stripe on "syb_tsm_obj3"  
stripe on "syb_tsm_obj4"
```

In this example, each stripe has a unique backup object name; Backup Server appends a stripe identifier to each one.

Backup Server prints:

```
Backup Server session id is: 13. Use this value when executing the  
'sp_volchanged' system stored procedure after fulfilling any volume change  
request from the Backup Server.  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj0::00'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj1::01'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj2::02'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj3::03'  
Backup Server: 4.132.1.1: Attempting to open byte stream device:  
'syb_tsm::obj4::04'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740D1A6 ' section number 1  
mounted on byte stream 'syb_tsm::obj1::01'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740D1A6 ' section number 1  
mounted on byte stream 'syb_tsm::obj0::00'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740D1A6 ' section number 1  
mounted on byte stream 'syb_tsm::obj3::03'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740D1A6 ' section number 1  
mounted on byte stream 'syb_tsm::obj4::04'  
Backup Server: 6.28.1.1: Dumpfile name 'testdb092740D1A6 ' section number 1  
mounted on byte stream 'syb_tsm::obj2::02'  
Backup Server: 4.188.1.1: Database testdb: 24 kilobytes (87%) DUMPED.  
Backup Server: 4.188.1.1: Database testdb: 106 kilobytes (88%) DUMPED.  
Backup Server: 4.188.1.1: Database testdb: 220 kilobytes (90%) DUMPED.  
Backup Server: 4.188.1.1: Database testdb: 334 kilobytes (92%) DUMPED.  
Backup Server: 4.188.1.1: Database testdb: 874 kilobytes (100%) DUMPED.  
Backup Server: 3.43.1.1: Dump phase number 1 completed.  
Backup Server: 3.43.1.1: Dump phase number 2 completed.  
Backup Server: 3.43.1.1: Dump phase number 3 completed.  
Backup Server: 4.188.1.1: Database testdb: 898 kilobytes (100%) DUMPED.  
Backup Server: 3.42.1.1: DUMP is complete (database testdb).
```

The Backup Server error log displays the TSM backup object name and its IDs.

```
Oct 1 14:54:33 2009: A00: Database 'testdb' dumped. Tivoli Storage Manager  
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj0.0,  
Copyids: (0, 80904).
```

```
Oct 1 14:54:33 2009: A03: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj3.3,
Copyids: (0, 80903).
Oct 1 14:54:33 2009: A02: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj2.2,
Copyids: (0, 80905).
Oct 1 14:54:33 2009: A01: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj1.1,
Copyids: (0, 80902).
Oct 1 14:54:33 2009: A04: Database 'testdb' dumped. Tivoli Storage Manager
backup object name: fs = /demo_srv1, high = /testdb.DB, low = /obj4.4,
Copyids: (0, 80906).
Oct 1 14:54:34 2009: Backup Server: 3.42.1.1: DUMP is complete (database
testdb).
```

To load this dump to the same database, enter:

```
load database testdb from "syb_tsm::obj0"
stripe on "syb_tsm::obj1"
stripe on "syb_tsm::obj2"
stripe on "syb_tsm::obj3"
stripe on "syb_tsm::obj4"
```

Listing a server's backup objects

Use the `sp_querysmobj` stored procedure to retrieve a list of a server's backup objects. See the *Reference Manual: Procedures* for complete syntax and usage information.

Deleting backup objects from TSM

Use the `sp_deletesmobj` stored procedure to delete some or all of the current server's backup objects from TSM. See the *Reference Manual: Procedures* for complete syntax and usage information.

Index

B

- backup system with load database syntax for 7
- backup system with load transaction syntax for 7
- backup system with TSM
 - configuring for different source and target machines 5
 - data organization 6
 - deleting backup objects 13
 - dump and load when databases are different 8
 - dump database syntax for 7
 - dump transaction syntax for 7
 - environment variables for 3
 - installing and setting up 2
 - listing backup objects 13
 - logical structures 6
 - object naming 6
 - syb_tsm keyword 7
 - sybtsmpasswd utility 4
 - using multiple stripes 10
 - when source and target servers are different 9

D

- data compression, configuring 5
- data hierarchy 7, 9
- deleting backup objects 13
- dump database command 7, 8, 10
- dump transaction command 7

L

- libsyb_tsm module 2
- listing backup objects 13
- load database command 7
- load transaction command 7
- logical structure, TSM 6

M

- multiple stripes, backups with 10

O

- object names 7

S

- sp_deletesmobj stored procedure 4, 13
- sp_querysmobj stored procedure 4, 13
- syb_tsm keyword 7
- sybtsmpasswd utility 4

T

- Tivoli Storage Manager (TSM) 1–13
 - and Adaptive Server 1
 - integration with Adaptive Server 1

