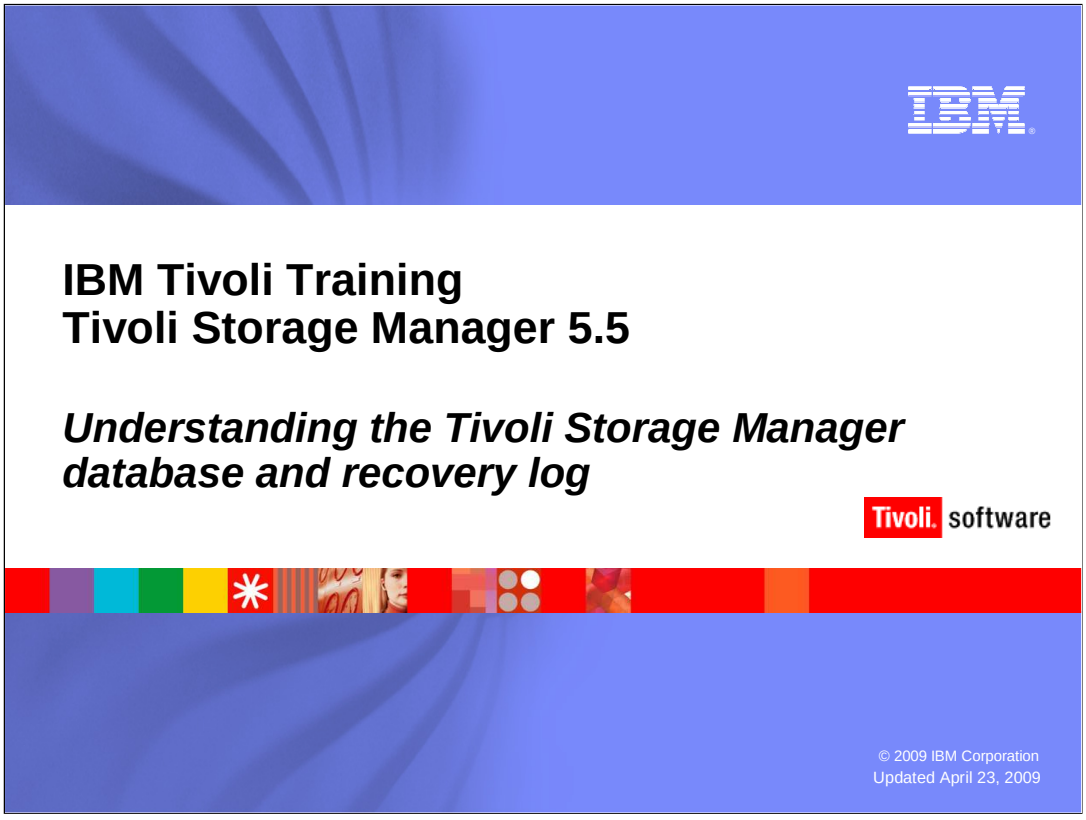


Slide 1

Welcome to the IBM Education Assistant training for IBM Tivoli Storage Manager version 5.5. This module will provide an explanation of how the TSM database and recovery log work to control data storage. Tips and tricks on TSM database and recover log configuration will also be presented.

Objectives

Upon completion of this module, you should be able to:

- Identify the purpose of the database and recovery log volumes.
- Choose the location of the database and recovery log.
- Determine the size of the database and recovery log.
- Explain how to configure the database and recovery log to optimize performance.

Slide 2 Objectives

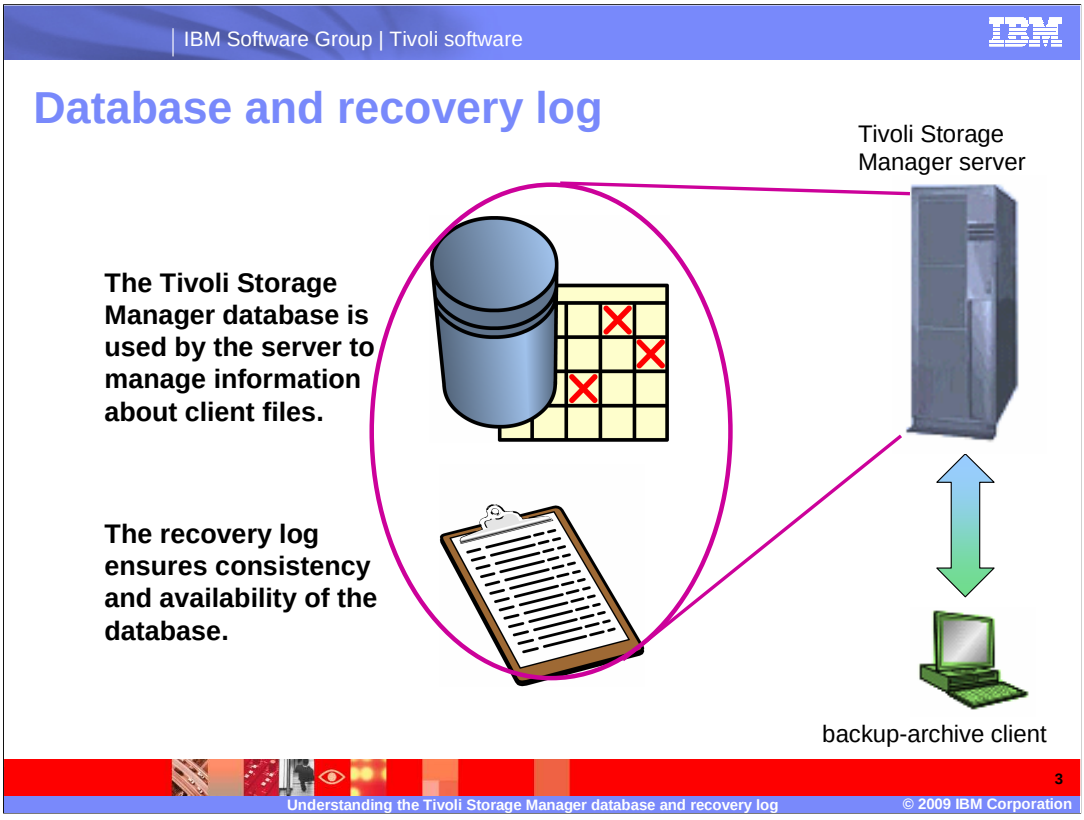
Upon completion of this module, you should be able to:

Identify the purpose of the database and recovery log volumes.

Choose the location of the database and recovery log.

Determine the size of the database and recovery log.

Explain how to configure the database and recovery log to optimize performance.



Slide 3 Database and recovery log

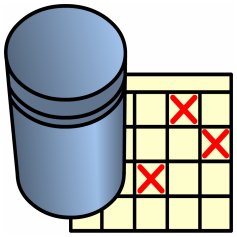
The Tivoli Storage Manager database contains information that is needed for server operations and information about client data that has been backed up, archived, and spacemanaged. The database does not store client data. Instead, the database points to the locations of the client files in the storage pools.

The recovery log contains information about database updates that have not yet been committed. Updates can include activities such as defining a management class, backing up a client file, and registering a client node. Changes to the database are recorded in the recovery log to maintain a consistent database image.

The Tivoli Storage Manager database

The Tivoli Storage Manager database contains server and client definition, including:

- Client nodes and administrators
- Policies and schedules
- Server settings
- Locations of client files on server storage
- Server operations



The information stored in the database is actually metadata, which describes the stored client data.

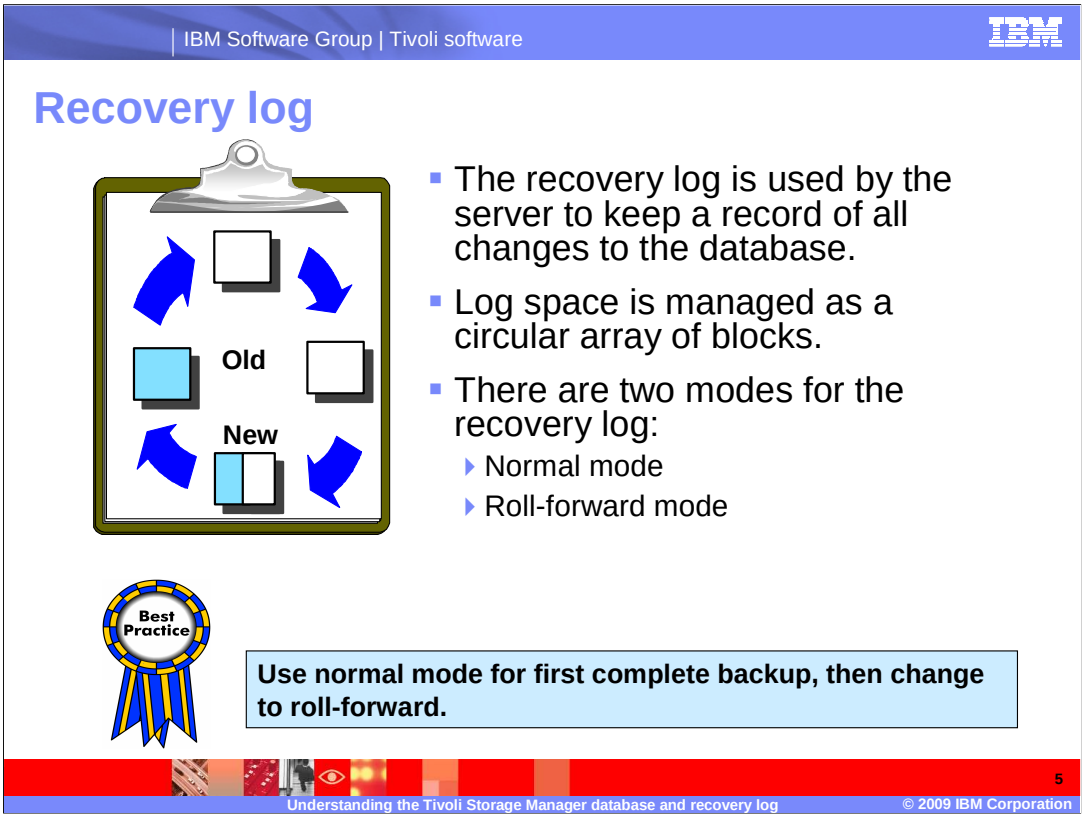
Slide 4 The Tivoli Storage Manager database

The Tivoli Storage Manager database contains all server and most client definitions. The TSM database does not contain client data.

The database includes information about:

- Client nodes and administrators
- Policies and schedules
- Server settings
- Locations of client files on server storage (not the actual client files)
- Server operations (for example, activity logs and event records)

To maintain the integrity of the data, the database uses a recovery log. The recovery log rolls back any changes made when a storage transaction is interrupted before it completes.



Slide 5 Recovery log

The recovery log is used by the server to keep a record of all changes to the database. When a change occurs, the recovery log is updated with some transaction information prior to the database being updated. This enables uncommitted transactions to be rolled back during recovery, so the database remains consistent.

The log is treated as a circular array of blocks with the head (the newest log records) always chasing the tail (oldest records). The server will never let the head overtake and overwrite the tail; it must take some other action.

There are two modes for the recovery log; normal mode and roll-forward mode.

Normal Mode

When the transaction log record is written to the recovery log, a recovery point is recorded in it, and the data is committed to the database. If a failure occurs before a transaction is committed to the database, the server rolls back any changes made to the database pages.

Roll-Forward Mode

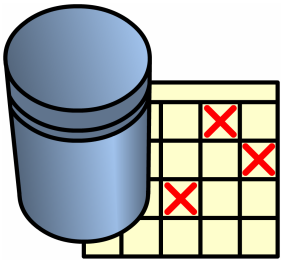
In roll-forward mode, all changes made to the database since the last database backup are saved in the recovery log. With roll-forward mode and an intact recovery log, you can recover the database up to its most current state (the point at which the database was lost).

TSM database space allocation

Considerations:

- The database is predominately read-oriented.
- The TSM database volumes should be distributed across disks for improved performance.

Refer to the *IBM Tivoli Storage Manager Performance Tuning Guide* for more information.



Slide 6 TSM Database space allocation

Database access is predominately read-oriented. Placing the database volumes over multiple physical volumes may improve performance because this allows the logical volume manager to spread the I/O load over more volumes. This is also true when mirroring is used because the logical volume manager schedules read operations to the least busy volume in a mirror set. However, keep the number of volumes reasonable (less than 12), to reduce space used for logical volume manager overhead. The maximum database size limit is 530 GB. It is recommended that you keep the size of the database at about 100GB for performance reasons.

If you notice a performance problem, consider adding another TSM server.

Recovery log space allocation

Considerations:

- The recovery log is predominately write-oriented.
- The recovery log should not be distributed over multiple disks.
- The log file maximum size limit is 13 GB.

Refer to the *IBM Tivoli Storage Manager Performance Tuning Guide* for more information.

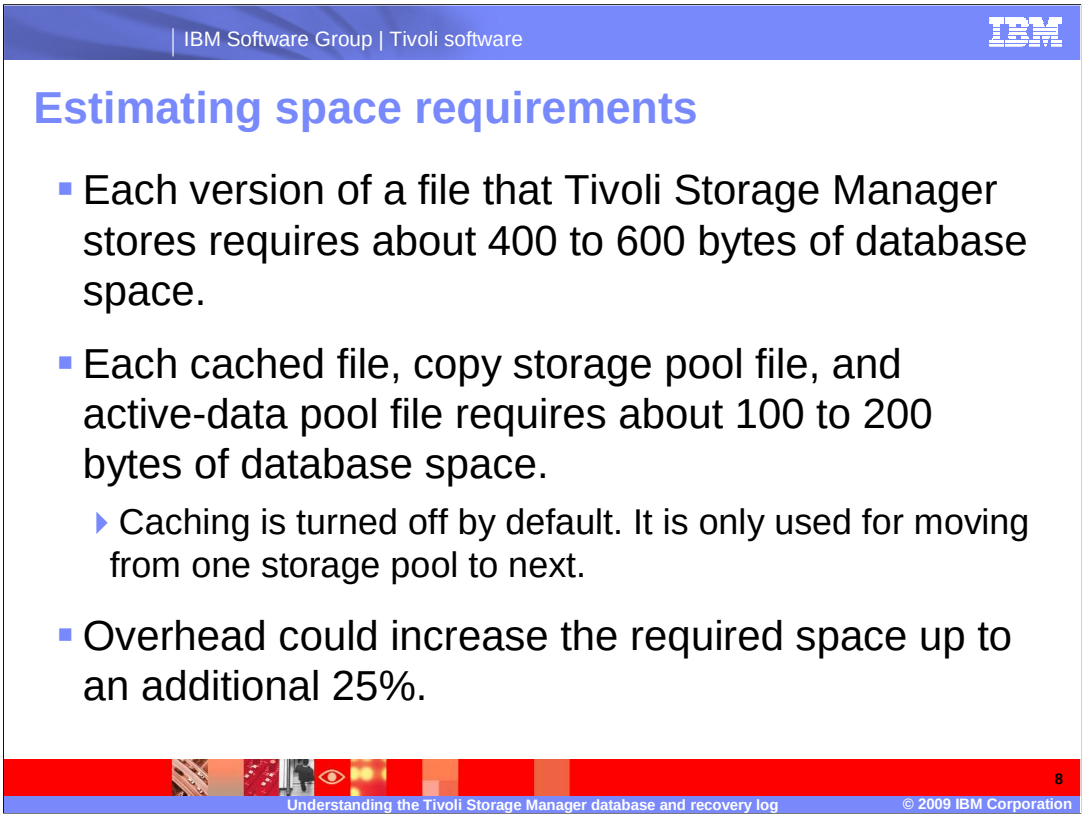
Allocate on a separate physical volume from the database.

Slide 7 Recovery log space allocation

Access to the recovery log is predominately write-oriented with the writes and the few reads clustered together. The writes are done in a moving cursor format, which does not lend itself to multiple volume optimizations. Therefore, fewer recovery log volumes are appropriate. Mirroring has little effect on the performance of the recovery log. The maximum log file size limit is 13 GB.

The size of the recovery log depends on the number of concurrent client sessions and the number of background processes executing on the server. The maximum number of concurrent client sessions is set in the server options. Begin with at least a 12MB recovery log. If you use the database backup and recovery functions in roll-forward mode, you should begin with at least a 25MB recovery log.

You can tune database and recovery log performance automatically or manually. You can set up triggers so that additional space is automatically added to the database and recovery log as needed. Volumes used to contain the database and the recovery log must be disk volumes.



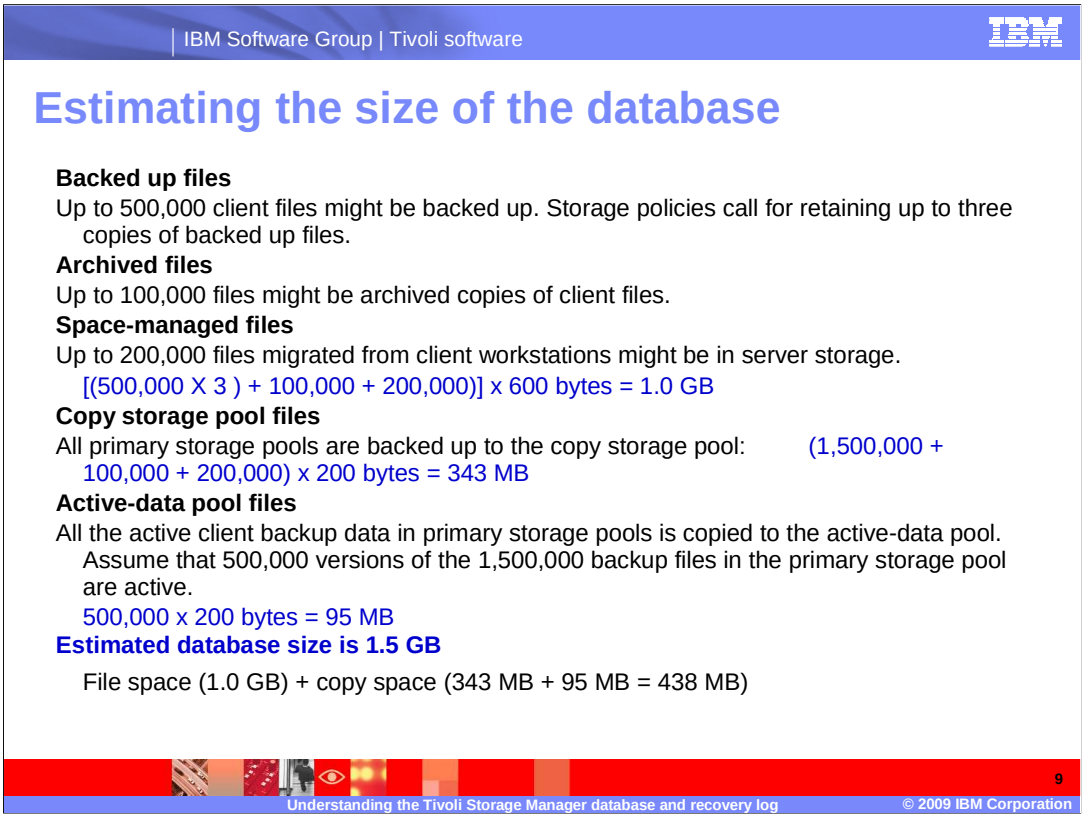
Slide 8 Estimating space requirements

The size of your Tivoli Storage Manager database depends on the number of client files to be stored and how Tivoli Storage Manager manages them. If you can estimate the maximum number of files that might be in server storage at any time, you can use the following information to come up with a useful database size estimate:

• Each version of a file that Tivoli Storage Manager stores requires about 400 to 600 bytes of database space.

• Each cached or copy storage pool copy of a file requires about 100 to 200 bytes of database space. Caching is turned off by default. It is only used for moving from one storage pool to next.

• Overhead could increase the required space up to an additional 25%.



Slide 9 Estimating the size of the database

In the following example, the computations are probable maximums.

The numbers are not based on the use of file aggregation. In general, small files that are aggregated require less database space.

Here are the specifics for this example:

Backed up files

Up to five hundred thousand client files might be backed up. Storage policies call for retaining up to three copies of backed up files.

Archived files

Up to 100,000 files might be archived copies of client files.

Space-managed files

Up 200,000 files migrated from client workstations might be in server storage.

So, three versions of 500,00 backed up files would be 1,500,000 client files. Add to that 100,000 archived files and 200,000 space managed files for a total 1,800,000 files. Multiple that times 600 bytes per file and you will need 1 GB of space for backed up, archived, and space managed files.

Copy storage pool files

All primary storage pools are backed up to the copy storage pool.

1,800,000 files times 200 bytes per file is 343 MB required for the copy storage pool files.

Active-data pool files

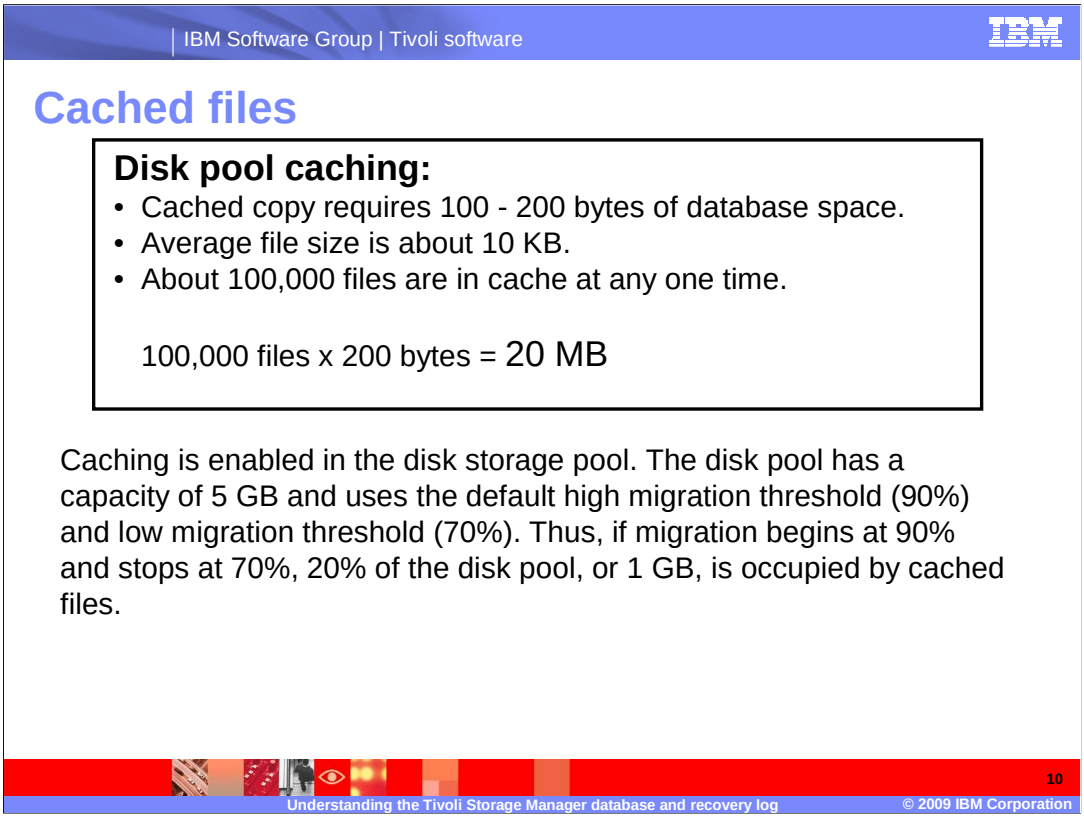
All the active client backup data in primary storage pools is copied to the active-data pool. Assume that 500,000 of the 1,500,000 backup files in the primary storage pool are active.

500,000 times 200 bytes per file is 95 MB required for the active-data pool.

Estimated database size is 1.5 GB

File space (1.0 GB) + copy space (343 MB + 95 MB = 438 MB)

Rounding up, you have 1.5 GB



Slide 10 Cached files

Cached copy requires 100 - 200 bytes of database space.

Average file size is about 10 KB.

About 100,000 files are in cache at any one time.

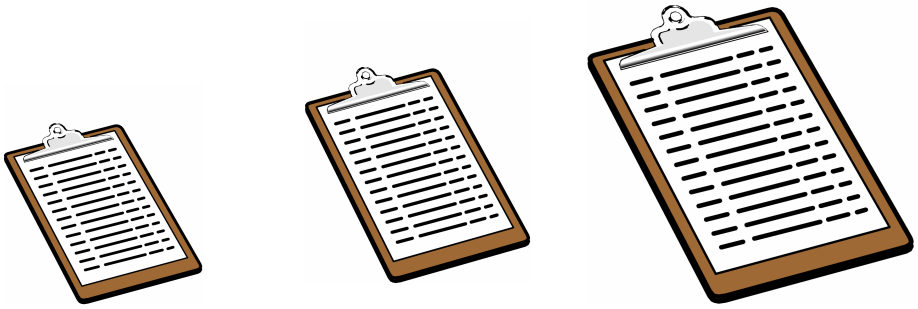
100,000 files x 200 bytes = 20 MB

Caching is enabled in the disk storage pool. The disk pool has a capacity of 5 GB and uses the default high migration threshold (90%) and low migration threshold (70%). Thus, if migration begins at 90% and stops at 70%, 20% of the disk pool, or 1 GB, is occupied by cached files.

Estimating the size of the recovery log

Easy estimation rules for recovery log size requirements:

(1 KB of recovery log space) x (daily number of transactions)
or
5% to 10% of total database size for normal mode
and
10% to 15% of total database size for roll-forward mode



Slide 11 Estimating the size of the recovery log

Easy estimation rules for recovery log size requirements:

(1 KB of recovery log space) x (daily number of transactions) or

5% to 10% of total database size for normal mode and

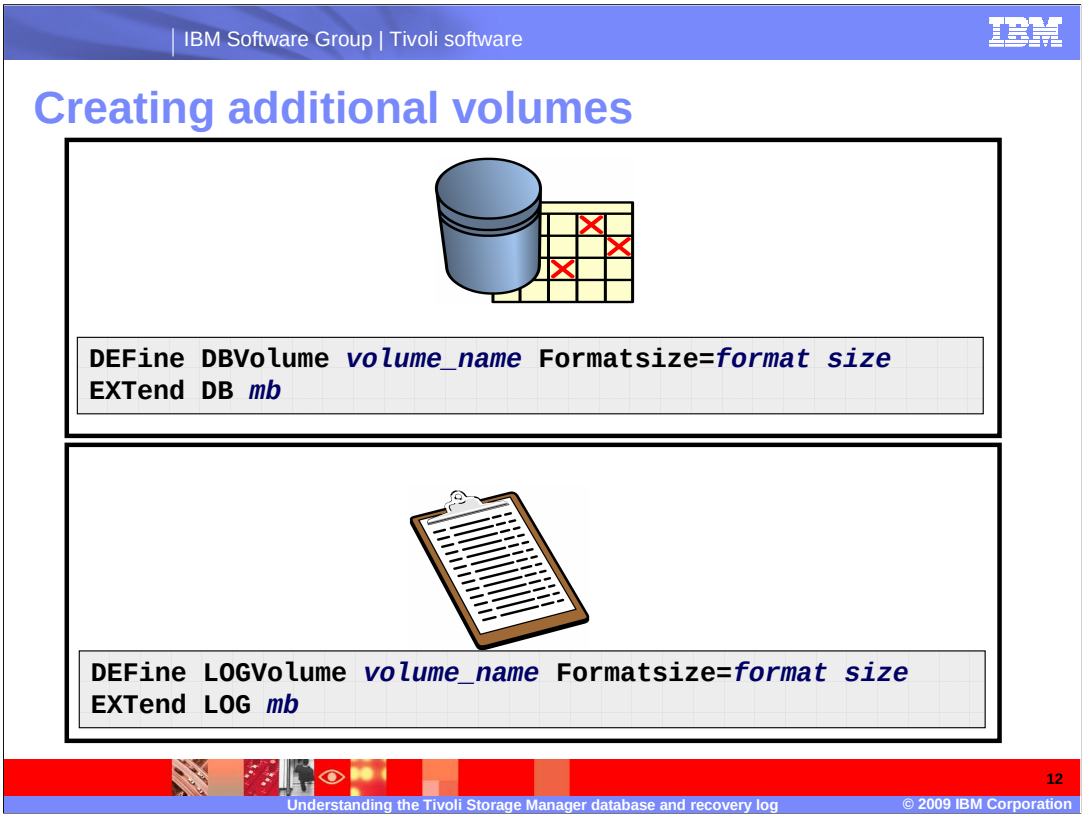
10% to 15% of total database size for roll-forward mode

In both normal mode and roll-forward mode, the volume of Tivoli Storage Manager transactions affects how large you should make your recovery log. As more clients are added and the volume of concurrent transactions increases, you should extend the size of the log. In roll-forward mode you must also consider how often you perform database backups. In this mode, the recovery log keeps all transactions since the last database backup and typically requires significantly more space than is required in normal mode.

In roll-forward mode, you need to determine how much recovery log space is used between database backups. For example, if you plan daily incremental backups, you should check your daily usage over a period of time.

You can use the following procedure to make your estimate.

For example, over a period of a week the highest cumulative consumption value was 500 MB. If you set your recovery log to 650 MB you should have sufficient space between daily backups.



Slide 12 Creating additional volumes

Define: Use the **define dbvolume** command to define a new database volume and use the **define logvolume** command to define a new recovery log volume.

Another method to allocate a new Tivoli Storage Manager database volume is by using the **define dbvolume** command with the **FORMATSIZE** option from either the Tivoli Storage Manager console or an administrative client.

Extend: Use the **extend db** command to increase the amount of space that can be used by the database within all the database volumes previously allocated to Tivoli Storage Manager. Use the **extend log** command to increase the amount of space that can be used by the recovery log within all the recovery log volumes previously allocated to Tivoli Storage Manager.

AIX example

DEFine DBVolume *volume_name* Formatsize=[format size]

`DEF DBVol /usr/tivoli/TSM/server/dbvol/db2.dsm F=12`

DEFine LOGVolume *volume_name* Formatsize=[format size]

`DEF LOGVol /usr/tivoli/TSM/server/rclog/log2.dsm F=8`

EXTend DB mb

`EXT db 12`

EXTend LOG mb

`EXT log 8`

The database name is case sensitive.

It is a best practice not to have your database and logs in the /bin directory.

Slide 13 AIX® example

The syntax for defining a database volume is:

DEFine DBVolume volume_name, which is the complete path, Formatsize=the size in megabytes. The database name is case sensitive.

The example: **DEF DBVol /usr/tivoli/TSM/server/dbvol/db2.dsm F=12**

The syntax for defining a log volume is:

DEFine LOGVolume volume_name Formatsize= the size in megabytes

The example: **DEF LOGVol /usr/tivoli/TSM/server/rclog/log2.dsm F=8**

The syntax for extending a database volume is:

EXTend DB size in megabytes

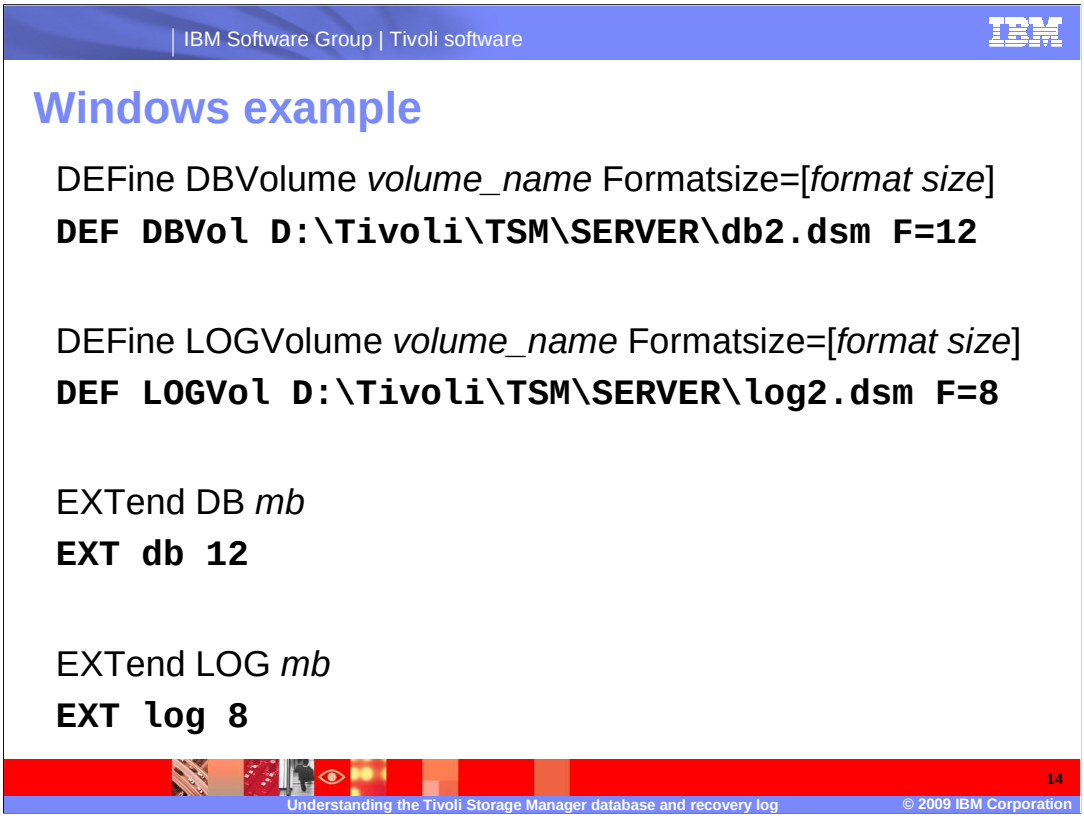
The example: **EXT db 12**

The syntax for extending a log volume is:

EXTend LOG size in megabytes

The example: **EXT log 8**

It is a best practice not to have your database and logs in the /bin directory.



Slide 14 Windows® example

You will notice that the syntax is the same, the volume name is different, due to the directory path structure.

The syntax for defining a database volume is:

DEFine DBVolume volume_name, which is the complete path, Formatsize=the size in megabytes

The example: **DEF DBVol D:\Tivoli\TSM\SERVER\db2.dsm F=12**

The syntax for defining a log volume is:

DEFine LOGVolume volume_name Formatsize= the size in megabytes

The example: **DEF LOGVol D:\Tivoli\TSM\SERVER\log2.dsm F=8**

The syntax for extending a database volume is:

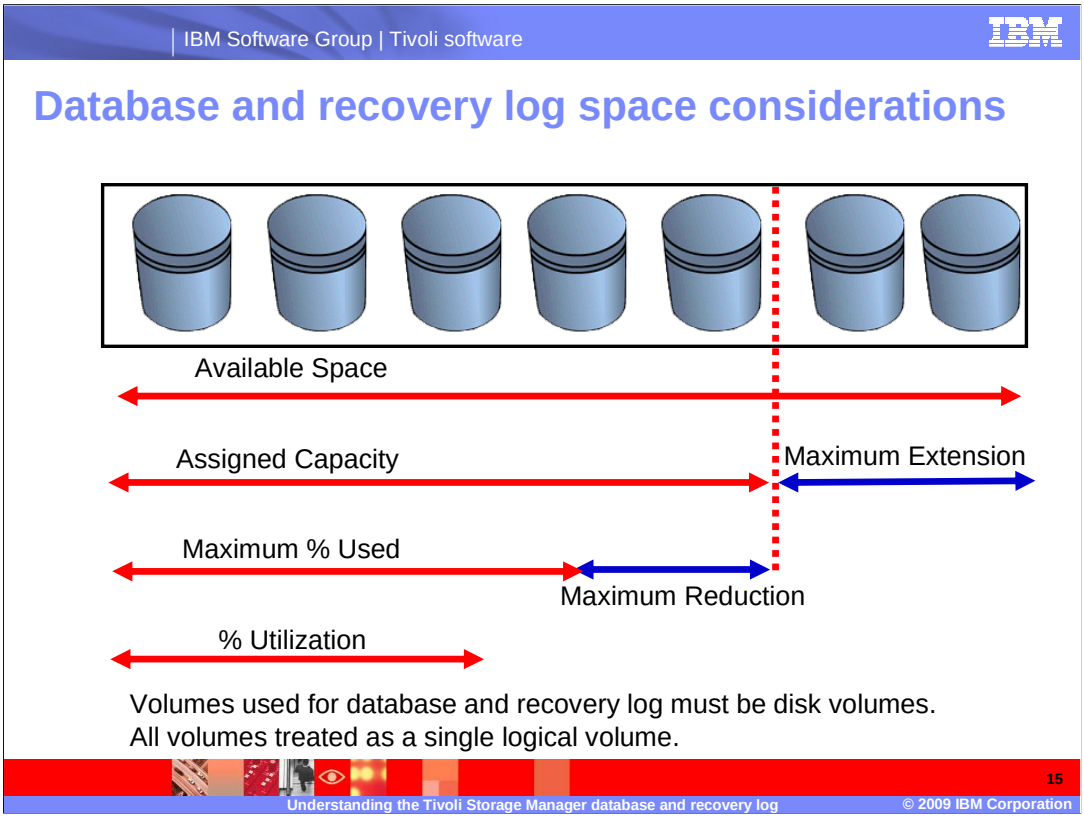
EXTend DB size in megabytes

The example: **EXT db 12**

And the syntax for extending a log volume is:

EXTend LOG size in megabytes

The example: **EXT log 8**



Slide 15 Database and recovery log space considerations

Remember, volumes used to contain the database and the recovery log must be disk volumes. Tivoli Storage Manager treats all volumes associated with the database or with the recovery log as a single logical volume. The logical volume manager maps data between logical and physical storage, allowing database and recovery log data to span physical disks. No reorganization of the database or recovery log is required.

The amount of space available for the database or recovery log equals the combined space of all volumes defined to the database or recovery log. As data is added, Tivoli Storage Manager tracks the percentage of utilization, which is the amount of space used at a specific point in time. Be aware that the maximum amount of space used by the recovery log can vary significantly throughout the day. It is proportional to the transaction load on the system. The maximum amount of space used by the database is more consistent with the utilization percentage, because the amount of database space consumed grows in proportion to the number of objects inserted into the database.

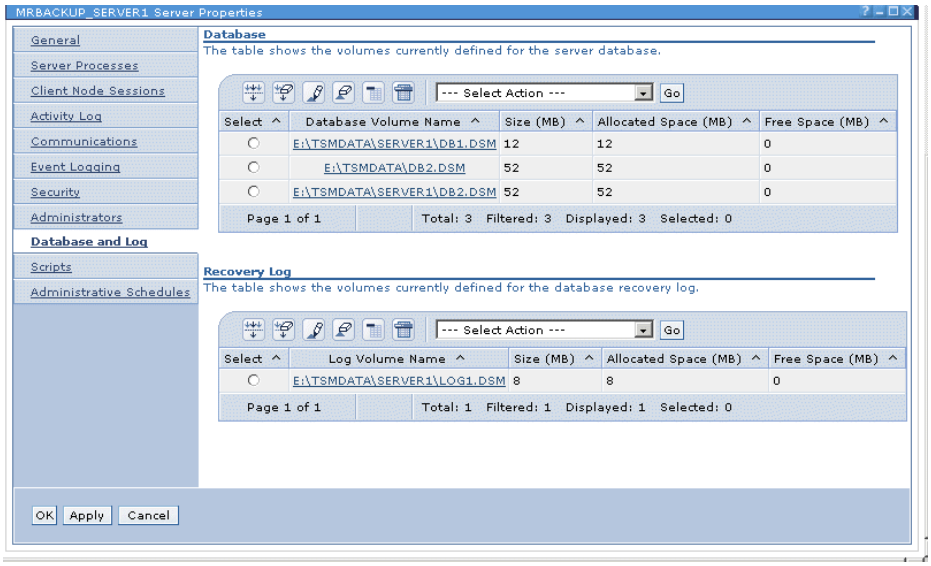
Note: It is a good practice to always have extension space available.

When you create a new volume, you can query the database or recovery log (**query db** and **query log** commands) to verify their assigned capacities. The value in the *Maximum Extension* field should equal the available space of the new volume. The *Maximum Reduction* field shows the assigned capacity not in use.

Extend–The database and recovery log are extended in 4MB increments. If you do not specify the extension in 4MB increments, the server rounds up to the next 4MB partition. For example, if you specify 1MB, the server extends the capacity by 4MB.

There are many ways to improve performance through strategic volume allocation. Although you can use a single, large Tivoli Storage Manager database, better performance can be achieved over time by using multiple volumes. This allows Tivoli Storage Manager to run more I/O requests to the database at any given time. If possible, these volumes should be placed either on different storage devices or on arrays for maximum performance. Some improvement can still be gained by placing multiple volumes on a single disk. However, performance improvements will not be as significant after four volumes are allocated per disk. Refer to the *Administrator's Guide* and the *Performance Tuning Guide* for more information.

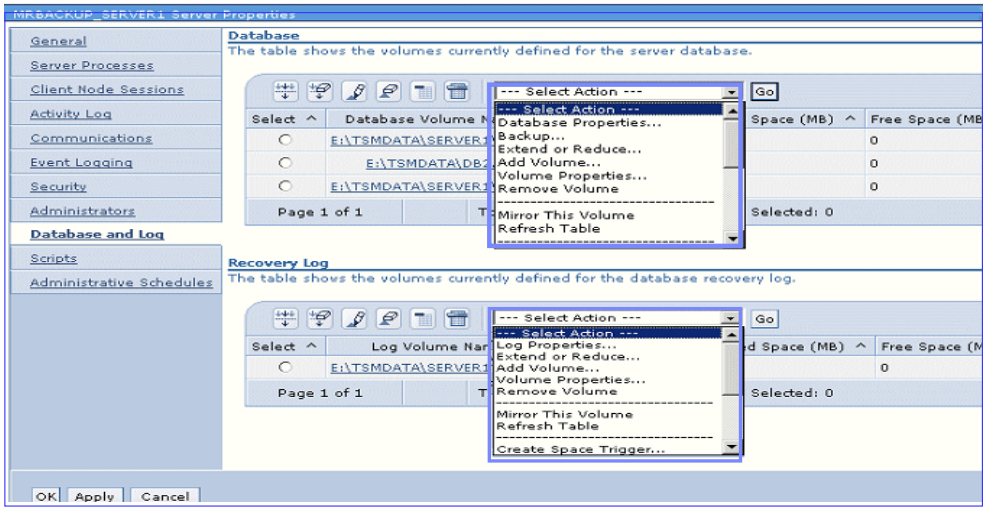
Using the Administration Center to add, view, and modify database and log volumes



Slide 16 Using the Administration Center to add, view, and modify database and log volumes

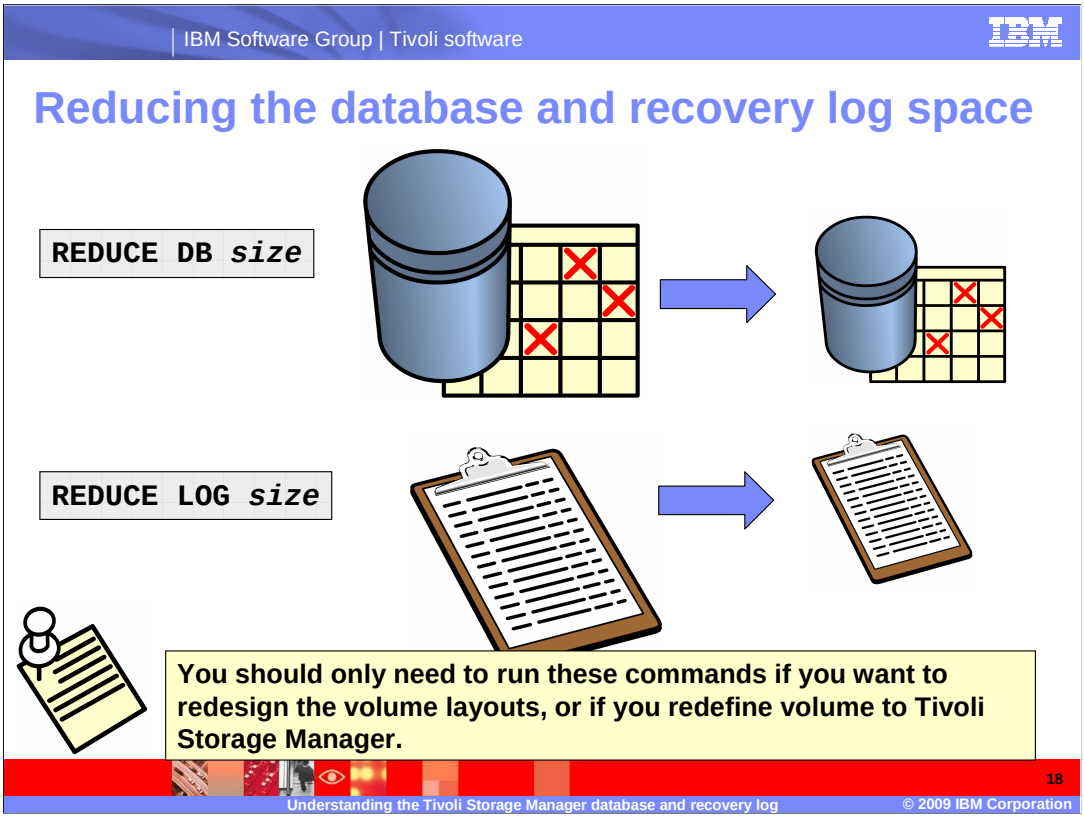
By selecting Database and Logs from the navigation tree in the left pane, you will see the database and log volumes. Use the Select Action pull-down menu to add a database or log volume. To View or modify an existing database or log volume, select the radio button next to the volume name.

The Administration Center's database and recovery log tasks



Slide 17 TSM Administration Center's database and recovery log tasks

Here you can see the actions you can select from the Select Action menu.



Slide 18 Reducing the database and recovery log space

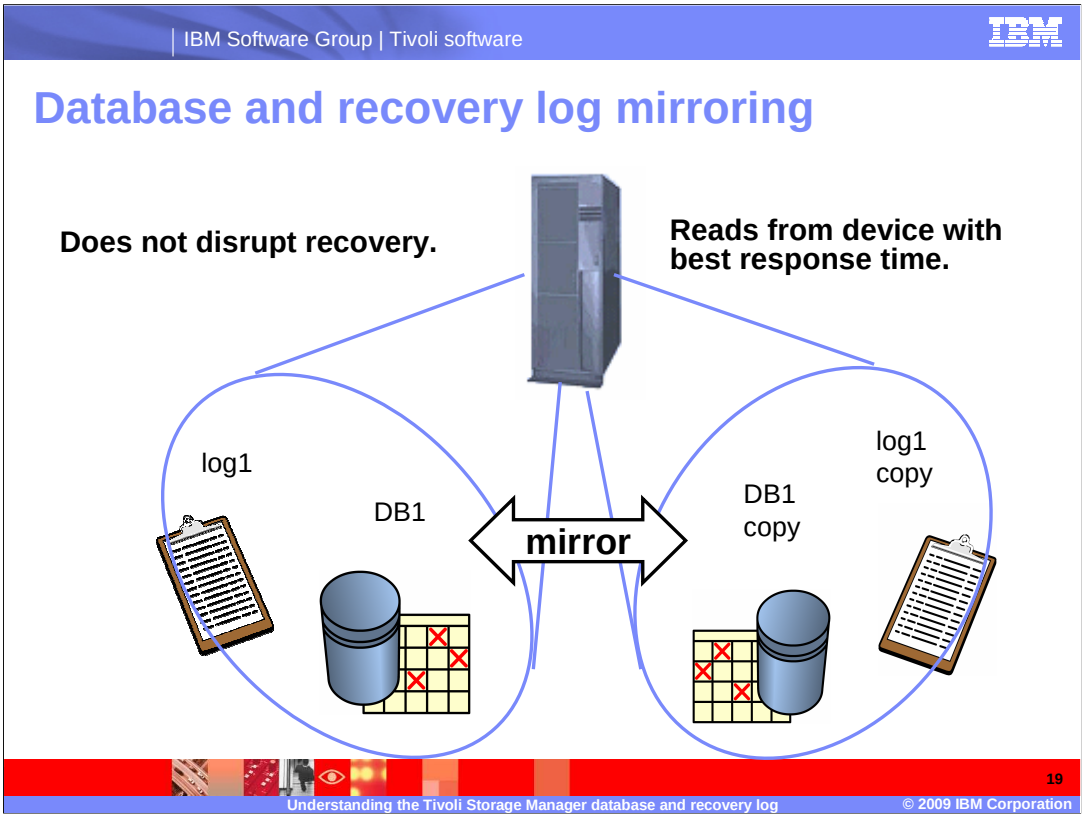
Use the **reduce db** command to decrease the amount of space that can be used by the database.

Use the **reduce log** command to decrease the amount of space that can be used by the recovery log. The log must be in normal mode.

When reducing the capacity of the database or recovery log, you must reduce by increments of 4MB. If you do not specify the reduction in 4MB increments, Tivoli Storage Manager rounds the number to the next 4MB partition.

In an example of reducing a database, you have 4 database volumes, DB1 through DB4. Based on the utilization of the database, you realize that DB1 alone *could* contain all the data. To reduce the database by the amount of available space in DB2, DB3, and DB4, which is 96MB, enter: **reduce db 96**. Reducing capacity is run as a background process and can take a long time. Issue a **query process** command to check on the status of the process.

Note: You should only need to run these commands if you want to redesign the volume or volume layout.



Slide 19 Database and recovery log mirroring

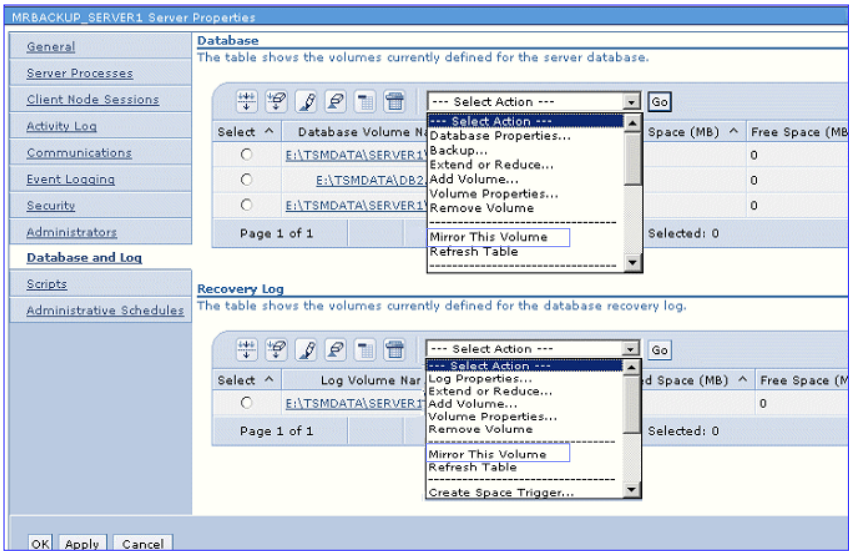
The availability of the database and recovery log is ensured by providing protection from logical errors and from device or media failures.

Mirroring offers additional flexibility to that of using a hardware dual copy. Mirroring has the following benefits:

- Three copies can be supported.
- You have finer control over what is mirrored.
- Copies may be on physical volumes that reside on different storage control units.
- There is support for multiple and different direct access storage device types.
- Read performance improves with mirroring because the server will be read from the device with the best response time.

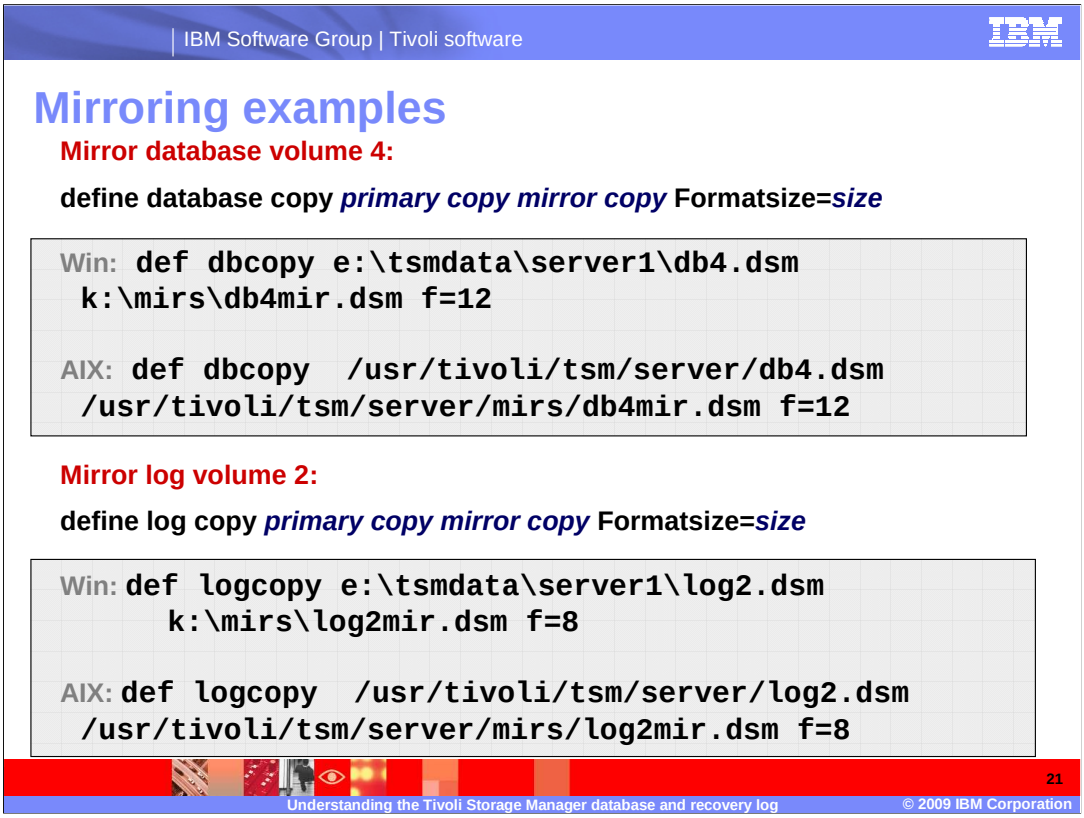
Note: It is not advisable to independently mirror volumes used for the database or recovery log, although it is allowed.

Using the Administration Center to mirror volumes



Slide 20 Using the Administration Center to mirror volumes

Mirror This Volume is an action you can select form the Select Action menu.



Slide 21 Mirroring examples

To format and define a database mirror, the syntax is:

define database copy primary copy mirror copy formatsize=size in MB

The Windows example to mirror database volume 4, where the primary copy is on the E drive and the mirror copy is on the K drive is:

def dbcopy e:\tsmdata\server1\db4.dsm (space) **k:\mirs\db4mir.dsm f=12**

The AIX example to mirror database volume 4:

def dbcopy /usr/tivoli/tsm/server/db4.dsm (space)
/usr/tivoli/tsm/server/mirs/db4mir.dsm f=12

To format and define a log mirror, the syntax is:

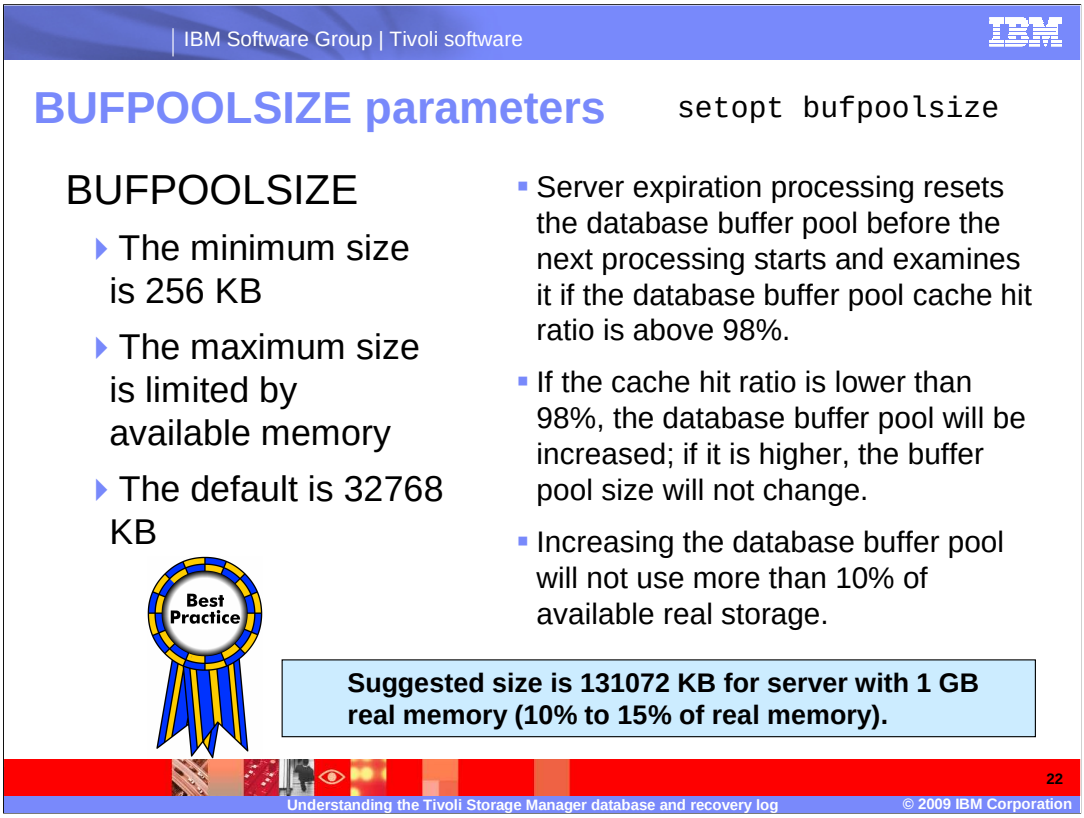
define log copy primary copy mirror copy Formatsize=size in MB

The Windows example to mirror log volume 2:

def logcopy e:\tsmdata\server1\log2.dsm (space) **k:\mirs\log2mir.dsm f=8**

The AIX example to mirror log volume 2:

def logcopy /usr/tivoli/tsm/server/log2.dsm (space)
/usr/tivoli/tsm/server/mirs/log2mir.dsm f=8



Slide 22 BUFPOOLSIZE parameters

Tivoli Storage Manager can dynamically adjust the size of the database buffer pool or you can adjust it manually.

The database buffer pool is storage that is used as a cache allowing database pages to remain in memory for long periods of time. This enables the server to make continuous updates to pages without requiring input or output operations from external storage.

To have the server automatically tune the **BUFPOOLSIZE** option, set the **SELFTUNEBUFPOOLSIZE** option to **YES**. The default is **NO**. If you specify **YES** for the **SELFTUNEBUFPOOLSIZE in the** server options, the database buffer pool is dynamically adjusted. The cache hit ratio statistics for the buffer pool are reset at the beginning of expiration. After expiration processing completes, the buffer pool size is adjusted dynamically.

Use the **setopt bufpoolsize** command to change the buffer pool size.

Server expiration processing resets the database buffer pool before the next processing starts and examines it if the database buffer pool cache hit ratio is above 98%.

If the cache hit ratio is lower than 98%, the database buffer pool will be increased; if it is higher, the buffer pool size will not change.

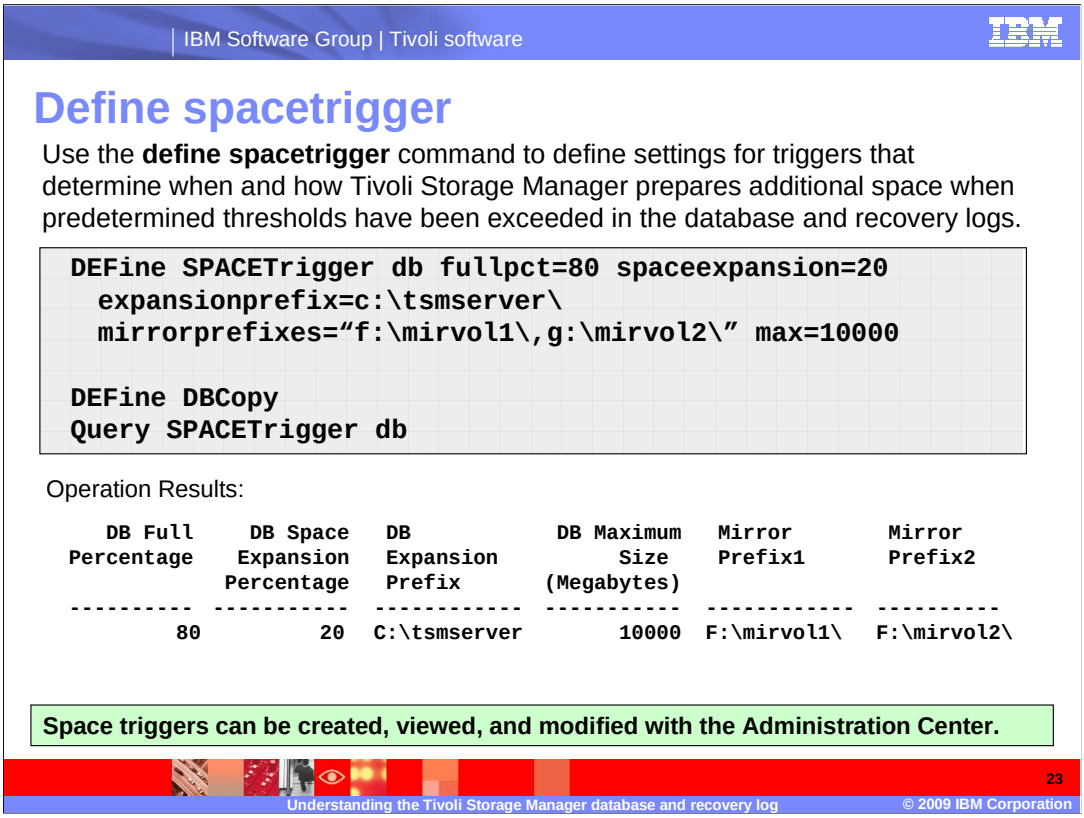
Increasing the database buffer pool will not use more than 10% of available real storage.

The minimum size is 256 KB

The maximum size is limited by available memory

The default is 32768 KB

Suggested size is 131072 KB for server with 1 GB real memory (10% to 15% of real memory).



Slide 23 Define spacetrigger

When the server performs an internal data movement operation, such as migration, reclamation, move data, or storage pool backup or restore, it will adjust the values of automatic performance tuning parameters, if configured, to achieve optimal performance.

To prevent running out of log space during these operations, use the **define spacetrigger** command to allow for expansion of the recovery log.

Administrator-defined space triggers tell the Tivoli Storage Manager server when to increase the size of the database or log. Space triggers define a maximum utilization percentage. When that utilization percentage is reached, the first step is to increase the size of the database or log, using space previously allocated but not used. The **extend** command is used to do this.

If there is no previously allocated space, or if there is insufficient space to reduce the utilization percentage below the trigger value, the database or log is expanded. The first step in expanding the database is to allocate a new volume. A volume prefix name and the amount of space by which to extend as a percentage of the existing database or log must be predefined. A constraining upper limit can be defined for the database or log to prevent extension beyond a certain point. When volume allocation formatting is finished, the database or log will be extended by the size of the newly allocated volumes. Mirrored volumes will be allocated as well if the database or log is mirrored.

The **define spacetrigger** command defines settings for the database and recovery log that determine when and how Tivoli Storage Manager deals with space shortages in the database and recovery log.

Space triggers can be created, viewed, and modified with the administration center.

Slide 24 **Training Roadmap for *IBM Tivoli Storage Manager***

If you go to www.ibm.com/software/tivoli/education/edu_prd.html this will take you to the training page, where you can access the training list for Tivoli Storage Manager version 5.5.

Summary

- You should now be able to:
  - Identify the purpose of the database and recovery log volumes.
  - Choose the location of the database and recovery log.
  - Determine the size of the database and recovery log.
  - Explain how to configure the database and recovery log to optimize performance.

Slide 25 Summary

You should now be able to:

Identify the purpose of the database and recovery log volumes.

Choose the location of the database and recovery log.

Determine the size of the database and recovery log.

Explain how to configure the database and recovery log to optimize performance.

Slide 26 (Copyright)

This concludes the IBM Education Assistant training for IBM Tivoli Storage Manager version 5.5, how the TSM database and recovery log work to control data storage**.**