B

# Performing Tablespace Point-in-Time Recovery



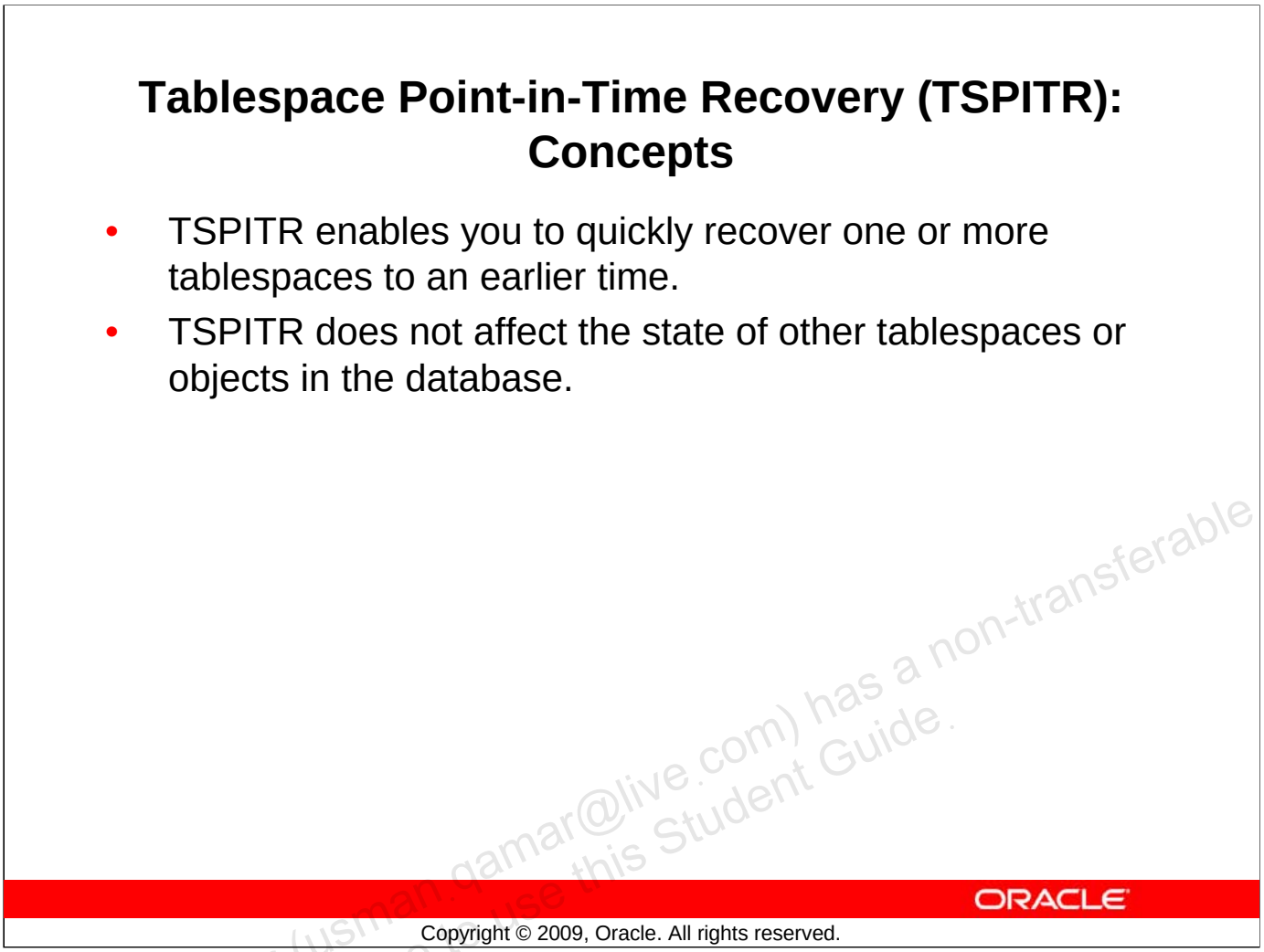
ORACLE

# Objectives

After completing this lesson, you should be able to:

- List what operations occur when you perform tablespace point-in-time recovery (TSPITR)
- Define the terminology used with TSPITR
- Identify the circumstances where TSPITR is a good solution
- Determine the correct target time for the point-in-time recovery
- Identify those situations where TSPITR cannot be used, and how to work around them
- Perform automated TSPITR

ORACLE

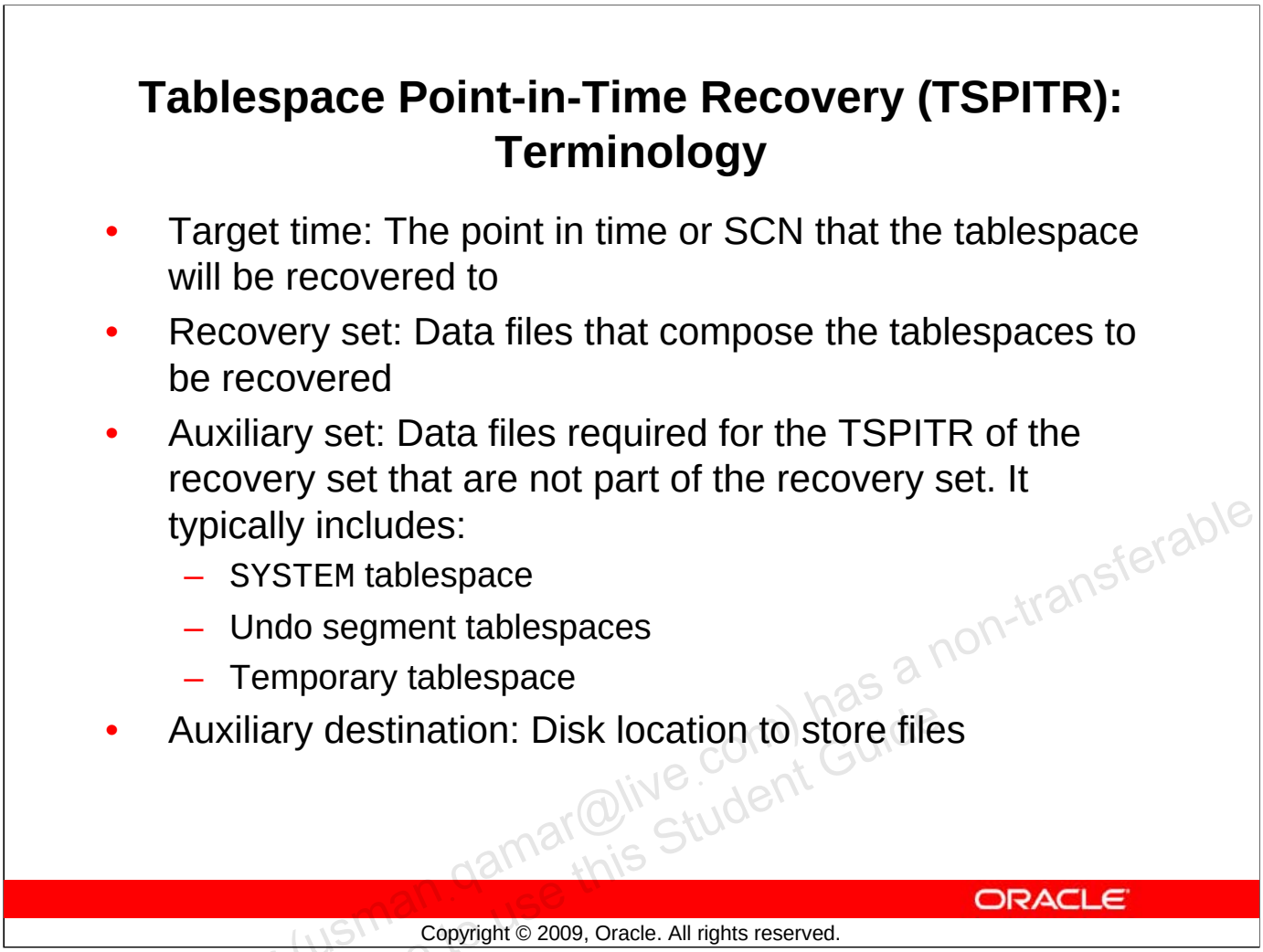







## Tablespace Point-in-Time Recovery (TSPITR): Concepts

RMAN automatic tablespace point-in-time recovery (TSPITR) enables you to quickly recover one or more tablespaces in an Oracle database to an earlier time, without affecting the state of the other tablespaces and objects in the database.





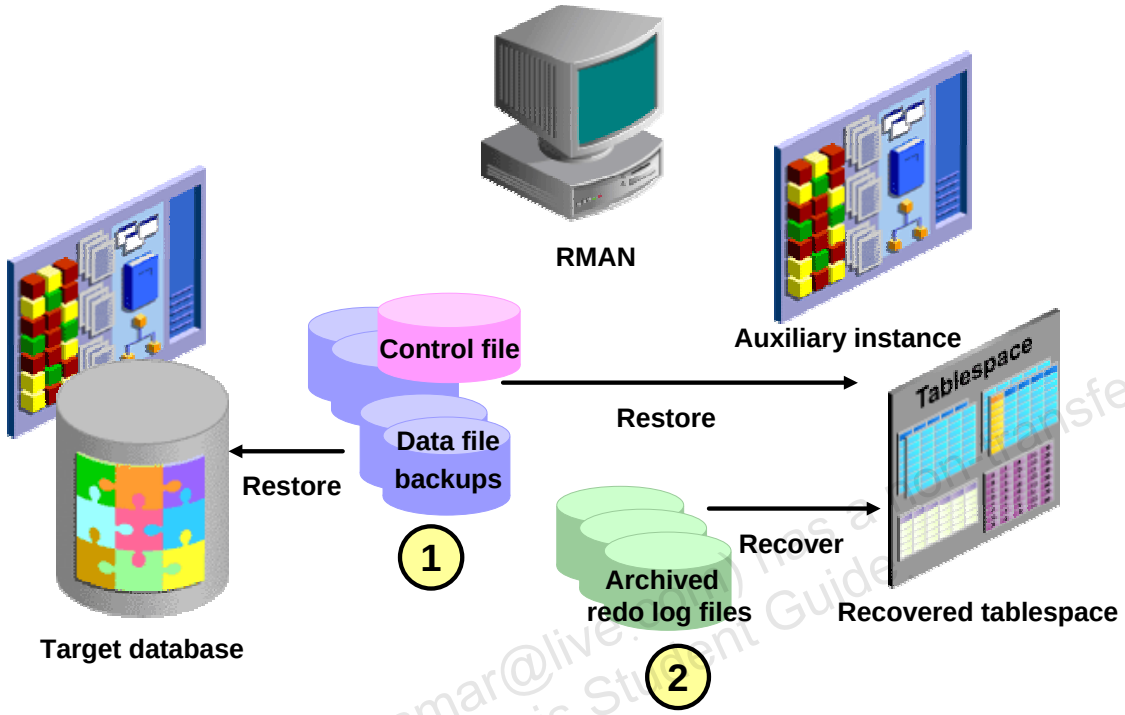



## Tablespace Point-in-Time Recovery (TSPITR): Terminology

The following terminology is used when discussing TSPITR:

- **Target time:** The point in time or system change number (SCN) that the tablespace will be recovered to during TSPITR
- **Recovery set:** Data files composing the tablespaces to be recovered
- **Auxiliary set:** Data files required for TSPITR of the recovery set that are not themselves part of the recovery set. The auxiliary set typically includes:
  - A copy of the SYSTEM tablespace
  - Data files containing undo segments from the target instance
  - In some cases, a temporary tablespace, used during the export of database objects from the auxiliary instance
- **Auxiliary destination:** A location on disk that can be used to store any of the auxiliary set data files, control files, and online logs of the auxiliary instance during TSPITR. Files stored in the auxiliary destination can be deleted after TSPITR is complete.

## Tablespace Point-in-Time Recovery: Architecture






### Tablespace Point-in-Time Recovery: Architecture

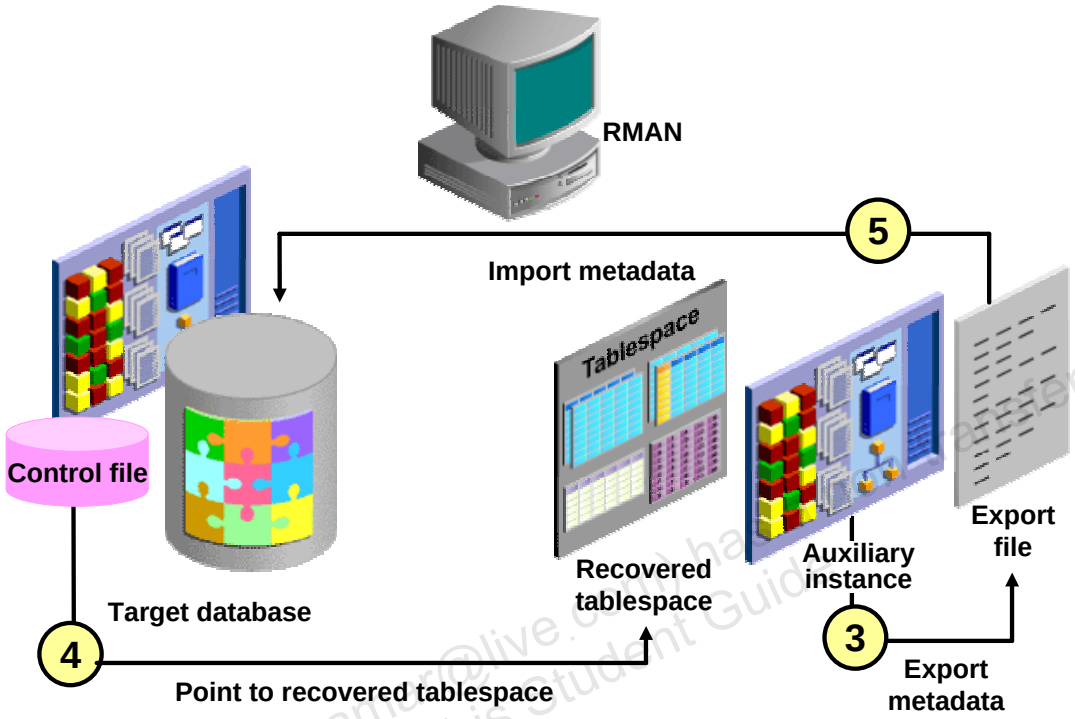
In the diagram, the following TSPITR entities are shown:

- **Target database:** Contains the tablespace to be recovered
- **Control file:** Provides backup information to RMAN
- **Backup sets:** Come from the target database and are the source of the reconstructed tablespace
- **Archived redo logs:** Come from the target database and are the source of the reconstructed tablespace
- **Auxiliary instance:** Is the Oracle database instance used during the recovery process to perform the recovery

RMAN performs the following steps during tablespace point-in-time recovery:

1. Restores a backup control file from a point in time before the target time to the auxiliary instance. It restores the data files for the *recovery set* to the target database and the data files for the *auxiliary set* to the auxiliary instance.
2. Recovers the restored data files to the specified point in time

## Tablespace Point-in-Time Recovery: Architecture






### Tablespace Point-in-Time Recovery Architecture (continued)

3. Exports the dictionary metadata about objects in the recovered tablespace to the target database
4. Issues `SWITCH` commands on the target database so that the target database control file points to the data files in the recovery set that were recovered on the auxiliary instance
5. Imports the dictionary metadata from the auxiliary instance to the target instance, allowing the recovered objects to be accessed

# When to Use TSPITR

- TSPITR can be used in the following situations:
  - To recover data lost after an erroneous TRUNCATE TABLE statement
  - To recover from logical corruption of a table
  - To undo the effects of a batch job or DML statements that have affected only a part of the database
  - To recover a logical schema to a different point from the rest of the physical database
- TSPITR uses transportable tablespaces and Data Pump, providing the following new capabilities and features:
  - TSPITR can be used to recover a dropped tablespace.
  - TSPITR can be performed repeatedly to points-in-time before the tablespace was brought online without requiring a recovery catalog.

ORACLE







## When to Use TSPITR

RMAN TSPITR can be used to:

- Recover data lost after an erroneous TRUNCATE TABLE statement
- Recover from logical corruption of a table
- Undo the effects of an incorrect batch job or another data manipulation language (DML) statement that has affected only a subset of the database
- Recover a logical schema to a different point in time than other parts of the physical database

Prior to Oracle Database 11*g* Release2, TSPITR used export and import for processing. TSPITR now uses transportable tablespaces and Data Pump. Because of this change to the underlying technology, TSPITR can be used to recover a dropped tablespace. In addition, TSPITR can be performed repeatedly to different points in time without the need for a recovery catalog.

## Preparing for TSPITR

To prepare for TSPITR, perform the following steps:

- Determine the correct target time.
- Determine what is needed in the recovery set.
- Identify and preserve objects that will be lost after TSPITR.



ORACLE



### Preparing for TSPITR

Before performing TSPITR, you need to determine the correct target time for your recovery. You need to determine whether you need additional tablespaces in your recovery set. You should evaluate what objects will be lost as a result of the TSPITR operation and determine how you want to preserve those objects.

Each of these steps is discussed in more detail in this lesson.

## Determining the Correct Target Time

- You cannot perform TSPITR a second time unless you are using a recovery catalog.
- After you perform TSPITR and bring the tablespace online, you cannot use a backup from an earlier time.
- Use the following methods to determine the correct target time:
  - Flashback Query
  - Flashback Transaction Query
  - Flashback Version Query
- Simple alternative to TSPITR: Flash back data (if still available as undo).



### Determining the Correct Target Time

It is extremely important that you choose the right target time or SCN for TSPITR. After you perform TSPITR and bring a tablespace online, you cannot use any backup from a time earlier than the moment you brought the tablespace online. In practice, this means that you cannot make a second attempt at TSPITR if you choose the wrong target time the first time, unless you are using a recovery catalog. However, if you have a recovery catalog, you can perform repeated TSPITR operations to different target times.

The current control file does not contain a record of an older incarnation of the recovered tablespace if you do not use a recovery catalog. Recovery with a current control file that involves the tablespace cannot use a backup taken prior to the time when you brought the tablespace online. However, you can perform incomplete recovery of the whole database to any time prior to or equal to the time when you brought the tablespace online if you can restore a backup control file from before that time.

You can use Oracle Flashback Query, Oracle Flashback Transaction Query, and Oracle Flashback Version Query to investigate changes to your database and to help determine the correct target time for TSPITR.

**Note:** With the Flashback tools and the data still available as undo data, it is usually much simpler to use the Flashback tools for undoing unwanted changes (rather than TSPITR).

# Determining the Tablespaces for the Recovery Set

- Use the TS_PITR_CHECK view to identify relationships that span recovery set boundaries.
- If objects in the tablespace that you are recovering have relationships with objects in other tablespaces, you can:
  - Add the tablespace that contains the related objects to the recovery set
  - Suspend the relationship for the duration of TSPITR
  - Remove the relationship
- Use the DBMS_TTS.TRANSPORT_SET_CHECK procedure to determine whether the tablespaces in the recovery set are self-contained.

ORACLE







## Determining the Tablespaces for the Recovery Set

If you are unsure whether you have accounted for all objects that have relationships with the objects in the tablespaces you are performing the TSPITR operation for, you can use the TS_PITR_CHECK view to identify any additional objects. When you query this view, information about any objects that will prevent you from proceeding with TSPITR is displayed. The reason why tablespace point-in-time recovery cannot proceed is displayed in the REASON column of the TS_PITR_CHECK view.

As an example, if you are planning to perform TSPITR for the USERS and EXAMPLE tablespaces, execute the following query to determine whether there are any relationships with objects in other tablespaces that are not accounted for:

```
SELECT * FROM SYS.TS_PITR_CHECK
WHERE (TS1_NAME IN ('USERS','EXAMPLE')
   AND TS2_NAME NOT IN ('USERS','EXAMPLE'))
OR (TS1_NAME NOT IN ('USERS','EXAMPLE')
AND TS2_NAME IN ('USERS','EXAMPLE'));
```

Refer to the *Oracle Database Backup and Recovery User's Guide* for additional examples using the TS_PITR_CHECK view.

# Identifying Relationships That Span Recovery Set Boundaries

```
SELECT *
FROM SYS.TS_PITR_CHECK
WHERE (
   TS1_NAME IN ('USERS','EXAMPLE')
   AND TS2_NAME NOT IN ('USERS','EXAMPLE'))
OR (
   TS1_NAME NOT IN ('USERS','EXAMPLE')
   AND TS2_NAME IN ('USERS','EXAMPLE'));
```

Use DBMS_TTS.TRANSPORT_SET_CHECK to ensure that TSPITR will be successful:

```
DBMS_TTS.TRANSPORT_SET_CHECK ('USERS', 'EXAMPLE');
SELECT *  FROM TRANSPORT_SET_VIOLATIONS;
```

ORACLE







### Identifying Relationships Between Objects That Span the Recovery Set Boundaries

Before performing TSPITR, you must determine the recovery set. If objects in the tablespaces you need to recover have relationships with objects in other tablespaces, you need to make provisions for those objects.

Prior to Oracle Database 11*g* Release 2, you used the SYS.TS_PITR_CHECK view to identify relationships between objects that span the recovery set boundaries. Now you should use the DBMS_TTS.TRANSPORT_SET_CHECK procedure and query the TRANSPORT_SET_VIOLATIONS view.

**Note:** RMAN TSPITR automatically executes the DBMS_TTS.TRANSPORT_SET_CHECK procedure for the recovery set tablespaces and verifies that the query against TRANSPORT_SET_VIOLATIONS returns no rows. If the query returns rows, RMAN stops TSPITR processing and any tablespace containment violations must be resolved before TSPITR can proceed. You can execute the procedure and query the view as described above as a precautionary measure.

# Identifying Objects That Will Be Lost

- Objects created in the tablespace after the target recovery time are lost.
- Query TS_PITR_OBJECTS_TO_BE_DROPPED to determine which objects will be lost following TSPITR.
- Use Export prior to TSPITR and Import following TSPTIR to preserve and re-create the lost objects.







### Identifying Objects That Will be Lost

Query the TS_PITR_OBJECTS_TO_BE_DROPPED view to determine whether there are any objects that will be lost as a result of performing tablespace point-in-time recovery.

As an example, you are performing TSPITR for the USERS and EXAMPLE tablespaces to the target time of April 3, 2006 at 8:30:00 AM. Issue the following query to determine whether there are any objects that will be lost following your TSPITR:

```
SELECT OWNER, NAME, TABLESPACE_NAME,
TO_CHAR(CREATION_TIME, 'YYYY-MM-DD:HH24:MI:SS')
FROM TS_PITR_OBJECTS_TO_BE_DROPPED
WHERE TABLESPACE_NAME IN ('USERS','EXAMPLE')
AND CREATION_TIME >
    TO_DATE('2006-APR-03:08:30:00','YY-MON-DD:HH24:MI:SS')
ORDER BY TABLESPACE_NAME, CREATION_TIME;
```

# Performing Basic RMAN TSPITR

- Fully automated TSPITR
  - Specify an auxiliary destination.
  - RMAN manages all aspects of TSPITR.
  - This is the recommended method.
- Customized TSPITR with an automatic auxiliary instance
  - This is based on fully automated TSPITR.
  - Customize the location of files.
  - Specify initialization parameters.
  - Specify channel configurations.
- TSPITR using your own auxiliary instance
  - Configure and manage the auxiliary instance.

ORACLE







## Performing Basic RMAN TSPITR

You have the following options when performing TSPITR:

- **Fully automated TSPITR:** Specify an auxiliary destination, and RMAN manages all aspects of the TSPITR operation. This is the simplest way to perform TSPITR, and is recommended unless you specifically need more control over the location of recovery set files after TSPITR or auxiliary set files during TSPITR, or control over the channel configurations or some other aspect of your auxiliary instance.
- **Customized TSPITR with an automatic auxiliary instance:** TSPITR is based on the behavior of fully automated TSPITR, possibly still using an auxiliary destination. You can customize one or more aspects of the behavior, such as the location of auxiliary set or recovery set files. You can specify initialization parameters or channel configurations for the auxiliary instance created and managed by RMAN.
- **TSPITR with your own auxiliary instance:** Set up, start, stop, and clean up the auxiliary instance used in TSPITR. In addition, you can manage the TSPITR process by using some of the methods available in customized TSPITR with an automatic auxiliary instance.

## Performing Fully Automated TSPITR

1. Configure channels required for TSPITR on the target instance.
2. Specify the auxiliary destination using the `AUXILIARY DESTINATION` option.

```
RMAN> CONNECT TARGET
RMAN> RECOVER TABLESPACE users, example
    > UNTIL TIME '2007-06-29:08:00:00'
    > AUXILIARY DESTINATION
    > '/u01/app/oracle/oradata/aux';
```

3. Back up the recovered tablespaces and bring them online.



### Performing Fully Automated TSPITR

In addition to the preparation requirements discussed earlier in the lesson, when you perform fully automated TSPITR, you must:

- Configure any channels required for TSPITR on the target instance
- Specify a destination for RMAN to use for the auxiliary set of data files and other auxiliary instance files

After TSPITR has completed, back up the recovered tablespaces and bring them online. You cannot use backups of any tablespaces that participate in TSPITR taken prior to TSPITR after you perform TSPITR.

**Note:** This time format assumes that `NLS_DATE_FORMAT` is set to `'yyyy-mm-dd:hh24:mi:ss'` and `NLS_LANG` is set to `AMERICAN_AMERICA.WE8MSWIN1252`.

# Using Image Copies for Faster TSPITR Performance

```
CONFIGURE AUXNAME FOR DATAFILE
    '$ORACLE_BASE/oradata/orcl/users01.dbf'
TO '/backup/users01.dbf';

RECOVER TABLESPACE users UNTIL SEQUENCE 1300 THREAD 1;
```

CONFIGURE AUXNAME is replaced by SET NEWNAME for recovery set data files:

```
RUN
{
SET NEWNAME FOR DATAFILE
    '$ORACLE_BASE/oradata/orcl/users01.dbf'
TO '/backup/users01.dbf';

RECOVER TABLESPACE users UNTIL SEQUENCE 1300 THREAD 1;
}
```

ORACLE







## Using Image Copies for Faster RMAN TSPITR Performance

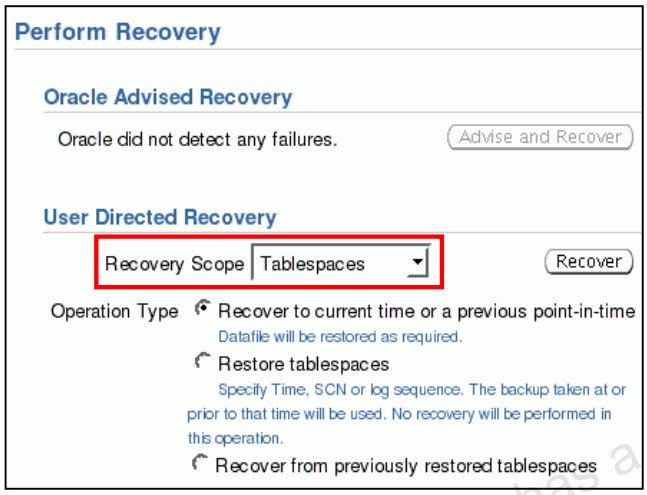
You can improve TSPITR performance by directing RMAN to use the existing image copies of the recovery set and auxiliary set data files. This technique enables RMAN to skip restoring the data files from a backup.

Prior to Oracle Database 11*g* Release 2, you used the following techniques to specify the existence of an image copy of a data file:

- CONFIGURE AUXNAME command for image copies of recovery set data files or auxiliary set data files
- SET NEWNAME command for image copies of auxiliary set data files

Now you should use the CONFIGURE AUXNAME command only with image copies of auxiliary set data files. You should use the SET NEWNAME command for image copies of recovery set data files or auxiliary set data files.

# Using Enterprise Manager to Perform TSPITR




ORACLE







## Using Enterprise Manager to Perform TSPITR

You can also use Enterprise Manager to perform TSPITR. Navigate to Availability > Perform Recovery. In the User Directed Recovery section, select Tablespaces from the Recovery Scope drop-down menu.

There are three operations you can perform, for tablespaces:

- **Recover to current time or a previous point in time:** Restores the data files for the tablespace, if needed. This operation then uses redo to recover to the time you specify: either the current time or a time in the past. This is the combination of the following two operations.
- **Restore tablespaces:** Only restore the data files for the tablespace. No recovery is performed.
- **Recover from previously restored tablespaces:** Perform recovery (redo application) only, of the tablespace's data files.

# RMAN TSPITR Processing

RMAN performs the following steps:

1. Creates the auxiliary instance, starts it, and connects to it
2. Takes the tablespaces that will be recovered offline
3. Restores a backup control file from a point in time before the target time to the auxiliary instance
4. Restores the data files from the recovery set and the auxiliary set to the auxiliary instance
5. Recovers the restored data files to the specified time
6. Opens the auxiliary database with the RESETLOGS option



ORACLE





**RMAN TSPITR Processing**

The steps that RMAN carries out to perform TSPITR are listed in this slide and the next.

# RMAN TSPITR Processing

7. Exports the dictionary metadata about objects in the recovered tablespaces to the target database
8. Shuts down the auxiliary instance
9. Imports the dictionary metadata from the auxiliary instance to the target instance
10. Deletes all auxiliary set files





## RMAN TSPITR Processing (continued)

After RMAN completes the last step, the TSPITR process is complete. The recovery set data files are returned to the state they were in at the specified target time.

## Performing RMAN TSPITR with an RMAN-Managed Auxiliary Instance

- Rename or relocate your recovery set data files.
- Specify a location other than the auxiliary destination for some or all of the auxiliary set data files.
- Create image copy backups of your data files in advance of TSPITR.
- Use a different channel configuration for the auxiliary instance.
- Specify different initialization parameters for your RMAN-managed auxiliary instance.







### Performing RMAN TSPITR with an RMAN-Managed Auxiliary Instance

If you want to customize RMAN TSPITR, you can use an RMAN-managed auxiliary instance and make the following changes:

- Rename the recovery set data files using `SET NEWNAME` so that they are not restored and recovered in their original locations.
- Control the location of your auxiliary set data files by specifying new names for individual files with `SET NEWNAME` and using `DB_FILE_NAME_CONVERT` to provide rules for converting data file names in the target database to data file names for the auxiliary database.
- Use existing image copies of the recovery set and auxiliary set data files on disk rather than restoring them from backup for faster RMAN TSPITR performance.

**Note:** Refer to the *Oracle Database Backup and Recovery User's Guide* for additional information.

## Performing RMAN TSPITR Using Your Own Auxiliary Instance

- Not recommended, but supported
- Perform the following steps:
  1. Create an Oracle password file for the auxiliary instance.
  2. Create an initialization parameter file for the auxiliary instance.
  3. Verify Oracle Net connectivity to the auxiliary instance.
  4. Start the auxiliary instance in `NOMOUNT` mode.
  5. Connect the RMAN client to the target and auxiliary instances.
  6. Execute the `RECOVER TABLESPACE` command.



ORACLE



### Performing RMAN TSPITR Using Your Own Auxiliary Instance

Oracle recommends that you allow RMAN to manage the creation and destruction of the auxiliary instance used during RMAN TSPITR. However, creating and using your own auxiliary instance is supported.

To create an Oracle instance suitable for use as an auxiliary instance, perform the following steps:

1. Create an Oracle password file for the auxiliary instance by using the `orapwd` utility.
2. Create a text initialization parameter file for the auxiliary instance.
3. Verify Oracle Net connectivity to the auxiliary instance using a valid net service name.

To perform TSPITR, complete the following steps:

4. Start the auxiliary instance in `NOMOUNT` mode.
5. Connect the RMAN client to target and auxiliary instances.
6. Execute the `RECOVER TABLESPACE` command.

Refer to the *Oracle Database Backup and Recovery User's Guide* for a detailed example.

# Troubleshooting RMAN TSPITR

- File name conflicts: Ensure that there are no name conflicts when using `SET NEWNAME`, `CONFIGURE AUXNAME`, and `DB_FILE_NAME_CONVERT`.
- RMAN cannot identify tablespaces with undo segments: Use the `UNDO TABLESPACE` clause.
- Restarting a manual auxiliary instance after TSPITR failure: Shut down and restart in `NOMOUNT` mode.



ORACLE





## Troubleshooting RMAN TSPITR

**File name conflicts:** If your use of `SET NEWNAME`, `CONFIGURE AUXNAME`, and `DB_FILE_NAME_CONVERT` cause multiple files in the auxiliary or recovery sets to have the same name, you receive an error during TSPITR. To correct the problem, specify different values for these parameters to eliminate the duplicate name.

RMAN cannot identify tablespaces with undo segments: During TSPITR, RMAN needs information about which tablespaces had undo segments at the TSPITR target time. This information is usually available in the recovery catalog, if one is used. If there is no recovery catalog, or if the information is not found in the recovery catalog, RMAN proceeds assuming that the set of tablespaces with undo segments at the target time is the same as the set of tablespaces with undo segments at the present time. If this assumption is not correct, the TSPITR operation fails and an error is reported. To prevent this from happening, provide a list of tablespaces with undo segments at the target time in the `UNDO TABLESPACE` clause.

Restarting manual auxiliary instance after TSPITR failure: If you are managing your own auxiliary instance and there is a failure in TSPITR, then before you can retry TSPITR, you must shut down the auxiliary instance, correct the problem, and put the auxiliary instance back in `NOMOUNT` mode.

# Summary

In this lesson, you should have learned how to:

- List what operations occur when you perform tablespace point-in-time recovery (TSPITR)
- Define the terminology used with TSPITR
- Identify the circumstances where TSPITR is a good solution
- Determine the correct target time for the point-in-time recovery
- Identify those situations where TSPITR cannot be used, and how to work around them
- Perform automated TSPITR