Phoenix backs up the transaction log, and you can use it to restore your database to a point-in-time. It is a useful, lightweight approach to backup and restore. Transaction log backups are incrementally triggered after a full backup job has completed, and a restore point exists. The transaction log is backed up every few minutes based on your settings. Phoenix uses the restore point, and the transactions that were logged, to perform a point-in-time restore. For more information on how the point-in-time restore works, see SQL Server Point in time restore workflow.

Phoenix allows you to restore databases to the original MS-SQL server location, or select another MS-SQL server instance. You can also restore the database to an AG. The restored database is replicated across the primary and secondary nodes of the AG.

Following sections provide information on how to:

- Restore the database to the original SQL server instance
- Restore a database an alternate SQL server instance
- Restore a database to an Availability Group

**Restore the database to the original SQL server instance**

Restore to the original server in case of an availability group (AG), restores the database to the instance of the primary server. When you restore a database to its original location, you can restore the database as a copy or you can replace the database on the SQL server instance with the restored database. If you replace the original database, Phoenix terminates connections to the database on the original instance and then replaces the database with the restored database.

**Restore the database as a copy**

1. Log in to the Phoenix Management Console.
2. On the menu bar, click All Organizations, and select the required organization from the drop-down list.
3. On the menu bar, click Protect > MS-SQL Servers.
4. On the SQL Backup Sets tab, click the resource name that hosts the backup set that you want to restore.
Tip: To quickly search and select a resource, you can use the search box on the SQL backup sets tab to filter the listed resources. Input the instance/AG name or the database name in the search tab to narrow down on the resource that you want to select.

5. On the server page, click **Restore > Point in Time Restore** on the backup set that you want to restore.
6. On the **Restore to point in the time** window, select **Restore to Original SQL Server instance** and click **Next**.
7. In the next section, select **Restore as a copy** and click **Next**.
8. In the **Select restore location** section, use one of the two options:
   - **Restore MDF, LDF, NDF files at different locations**: Enable this check box to restore the MDF, LDF, and NDF files at different locations. The fields that let you type or select the absolute path to the restore location appear when you enable this check box.
   - **Restore Location**: Type or select the absolute path to the restore location in this field. The database files are restored to the location you specify. This field disappears if you enable the above option.
   Click **Next**.
9. In the **Select time to restore databases** section, provide the following details:
   - Select a date and time based on the transactions that were backed up and the database mode after restore. Phoenix provides the following options for the database mode after restore.
   - **Under After restore, leave databases in**, select one of the following:
     - **Recovery mode**: In this mode, Phoenix rolls back all the uncommitted transactions and keeps the database in the ready state. Phoenix also ensures that all the data sets restored are consistent with the database and reports errors in case of inconsistencies.
     - **No recovery mode**: In this mode, Phoenix does not roll back any uncommitted transactions. The database is not usable in this intermediate, non-recovered state. This mode is used for restoring a database backup and multiple transaction logs, or when multiple RESTORE statements are needed.
     - **Standby mode**: In this mode, Phoenix allows a database to be brought up for the read-only access between the transaction log restores. This mode is used in the warm standby server situations or special recovery situations to inspect the database between the log restores. If you select this mode:
       - Choose the date and time of the logs available for download.
       - Provide the full folder path of the location where you want to restore additional transaction logs in the

https://docs.druva.com/Phoenix/030_Configure_Phoenix_for_Backup/050_Backup_and_Restore_MS_SQL_Server_database…
Updated: Tue, 15 Oct 2019 15:49:01 GMT
Powered by 2
Specify Location box.

After the restore job completes, Phoenix leaves the database in the selected mode.

Click Next.

10. Select the database to restore and click Finish.

Replace the database on the original SQL server instance

1. Log in to the Phoenix Management Console.
2. On the menu bar, click All Organizations, and select the required organization from the drop-down list.
3. On the menu bar, click Protect > MS-SQL Servers.
4. On the SQL Backup Sets tab, click on the resource name you want to restore.

Tip: To quickly search and select a resource, you can use the search box on the SQL backup sets tab to filter the listed resources. Input the instance/AG name or the database name in the search tab to narrow down on the resource that you want to select.

5. On the MS-SQL page, click Restore > Point in Time Restore.
6. On the Restore to point in the time window, select Restore to Original SQL Server instance and then click Next.
7. In the next section, select Replace original database and click Next.
8. Read the text in the confirmation dialog and click Yes to continue. If you do not want to proceed, click No and then you can choose to restore the database as a copy.
9. In the Select time to restore databases section, provide the following details:
   - Select a date and time based on the transactions that were backed up and the database mode after restore. Phoenix provides the following options for the database mode after restore.
   - Under After restore, leave databases in, select one of the following:
     - Recovery mode: In this mode, Phoenix rolls back all the uncommitted transactions and keeps the database in the ready state. Phoenix also ensures that all the data sets restored are consistent with the database and reports errors in case of inconsistencies.
- **No recovery mode**: In this mode, Phoenix does not roll back any uncommitted transactions. The database is not usable in this intermediate, non-recovered state. This mode is used for restoring a database backup and multiple transaction logs, or when multiple RESTORE statements are needed.

- **Standby mode**: In this mode, Phoenix allows a database to be brought up for the read-only access between the transaction log restores. This mode is used in the warm standby server situations or special recovery situations to inspect the database between the log restores. If you select this mode:
  - Choose the date and time of the logs available for download.
  - Provide the full folder path of the location where you want to restore additional transaction logs in the Specify Location box.

After the restore job completes, Phoenix leaves the database in the selected mode.

Click **Next**.

10. Select the database to restore and click **Finish**.

**Restore to alternate SQL server instance**

1. Log in to the Phoenix Management Console.
2. On the menu bar, click **All Organizations**, and select the required organization from the drop-down list.
3. On the menu bar, click **Protect > MS-SQL Servers**.
4. On the SQL Backup Sets tab, click the resource name that hosts the backup set that you want to restore.

**Tip**: To quickly search and select a resource, you can use the search box on the SQL backup sets tab to filter the listed resources. Input the instance/AG name or the database name in the search tab to narrow down on the resource that you want to select.

5. On the server page, click **Restore > Point in Time Restore** on the backup set that you want to restore.
6. On the Restore to point in the time window, select **Restore to alternate SQL server instance** and click **Next**.
7. In the Restore to alternate SQL Server Instance section, enter the details in the following fields:
   - **Restore MDF, LDF, NDF files at different locations**: Enable this check box to restore the MDF, LDF, and NDF files at different locations. The fields that let you type or select the absolute path to the restore location
appear when you enable this check box.

- **Select server**: From the Select server list, select the MS-SQL server to which you want to restore the database.
- **Select instance**: From the Select instance list, select the instance of the selected MS-SQL server to which you want to restore the database.
- **Specify location**: In the Specify Location box, type or select the absolute path to the restore location. The database files are restored to the location you specify. This field disappears if you enable the **Restore MDF, LDF, NDF files at different locations** check box.

Click **Next**.

8. In the **Select time to restore databases** section, provide the following details:

- Select a date and time based on the transactions that were backed up and the database mode after restore. Phoenix provides the following options for the database mode after restore.
- Under **After restore, leave databases in**, select one of the following:
  - **Recovery mode**: In this mode, Phoenix rolls back all the uncommitted transactions and keeps the database in the ready state. Phoenix also ensures that all the data sets restored are consistent with the database and reports errors in case of inconsistencies.
  - **No recovery mode**: In this mode, Phoenix does not roll back any uncommitted transactions. The database is not usable in this intermediate, non-recovered state. This mode is used for restoring a database backup and multiple transaction logs, or when multiple RESTORE statements are needed.
  - **Standby mode**: In this mode, Phoenix allows a database to be brought up for the read-only access between the transaction log restores. This mode is used in the warm standby server situations or special recovery situations to inspect the database between the log restores. If you select this mode:
    - Choose the date and time of the logs available for download.
    - Provide the full folder path of the location where you want to restore additional transaction logs in the Specify Location box.

After the restore job completes, Phoenix leaves the database in the selected mode.

Click **Next**.

9. Select the database to restore and click **Finish**.

**Restore to Availability Group**

Use Phoenix to add the restored database(s) to an AG. The restored databases are replicated across all the primary and secondary nodes in the AG. To use this feature, upgrade the Phoenix client to version 4.7.6 or later.

1. Log in to the Phoenix Management Console.
2. On the menu bar, click **All Organizations**, and select the required organization from the drop-down list.
3. On the menu bar, click **Protect > MS-SQL Servers**.
4. On the **SQL Backup Sets** tab, click the resource name that hosts the backup set that you want to restore.
Tip: To quickly search and select a resource, you can use the search box on the SQL backup sets tab to filter the listed resources. Input the instance/AG name or the database name in the search tab to narrow down on the resource that you want to select.

5. On the server page, click **Restore > Point in Time Restore** on the backup set that you want to restore.

6. On the **Restore to point in the time** window, select **Restore to Availability Group** and click **Next**.

7. In the **Restore to availability group** section, provide the following details:
   - **Restore MDF, LDF, NDF files at different locations**: Enable this check box to restore the MDF, LDF, and NDF files at different locations. The fields that let you type or select the absolute path to the restore location appear when you enable this check box.
   - **Select Availability Group**: From the **Select Availability Group** list, select the AG to restore the database. A database cannot be restored to an AG if there is no AG configured or the configured AG has an older client version.
   - **Restore Location**: In the **Restore Location** box, type or select the absolute path to the restore location. The database files are restored to the location you specify. This field disappears if you enable the **Restore MDF, LDF, NDF files at different locations** check box.
   - **Restore with automatic seeding**: Enable the **Restore with automatic seeding** check box to automatically replicate the restored databases to all the nodes in the AG. When you select this check box, the **Shared Network Location** field disappears from the page. You can use this option for SQL Server 2016 and later versions. For SQL Server 2012 and SQL Server 2014, you must manually replicate the database to all the nodes in the AG by specifying the shared network location.
   - **Shared Network Location**: Type or select the absolute path to the network location to synchronize the database across the primary and secondary nodes in the AG. Ensure that the shared network location is accessible to all the nodes in the AG. The **Shared Network Location** is a required field for SQL Server 2012 and SQL Server 2014, whereas it is an optional field for SQL Server 2016. This field disappears if the **Restore with automatic seeding** check box is enabled.

Click **Next**.

8. In the **Select time to restore databases** section, select a date and time based on the transactions that were backed up and click **Next**.
   
The **Databases** section lists the databases that you can restore to the specified date and timestamp.

9. Select the database to restore and click **Finish**.
Restoring database to a point-in-time with the standby mode enabled

When you restore the database using a transaction log, Phoenix provides an option to leave the database in the standby mode (step 3 of the above procedure). If you select the option to leave the database in the standby mode, you can restore your database to another point-in-time using transaction logs.

For example, when you are on step 3, you are notified that you can restore your database to any point between August 05, 2018, 01:48:43 PM and August 06, 2018 07:29:30 PM and you specify August 06, 2018, 07:29:30 PM as the time to restore your database. To restore the database:

- Phoenix identifies and selects the transaction closest to August 06, 2018, 07:29:30 PM.
- Transaction identified as closest to the time you specify can be the transaction logged right before the time you specify or, the transaction that was logged at the exact specified time.

If the option Leave database in Standby mode to allow restore of transaction logs is enabled,

- You can specify another date and time to restore your database to a transaction that occurred after August 06, 2018, 07:29:30 PM.
• The transaction logs that are logged after the specified date and time for restore are downloaded. You can use the downloaded logs to restore your database to a point in time.

In step 3, if you entered August 06, 2018, 07:29:30 PM, and August 06, 2018, 07:20:00 PM is the transaction closest to it,

• Your database is restored to the transaction on August 06, 2018, 07:20:00 PM, and
• You have downloaded logs that you can use to restore your database up to August 06, 2018, 07:29:30 PM.

Use the following command to apply logs:

Restore log [DB_name] from disk='log location' with standby='path_to_standy_file', stopat='timestamp'

Where the timestamp format is: 2018-08-06 07-29-30.

If the standby mode is not enabled:

◦ Your database is restored to the transaction that was logged closest to the time you specified.
◦ Your database is made operational.
◦ All the subsequent transaction logs are deleted, and there is no way you can restore your database to a transaction that was logged after the time you specified. In this example, you cannot restore your database to a transaction that was logged after August 06, 2018, 07:29:30 PM.

Point-in-time restore workflow

The following table depicts the point-in-time restore workflow using transaction logs:

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>You initiate a point-in-time restore.</td>
</tr>
<tr>
<td></td>
<td>Phoenix checks if the Phoenix agent is running.</td>
</tr>
<tr>
<td>2</td>
<td>If the Phoenix agent is running, Phoenix runs the restore job.</td>
</tr>
<tr>
<td></td>
<td>• If the Phoenix agent is not running, Phoenix queues the restore job. The restore job is completed after the Phoenix agent starts running.</td>
</tr>
<tr>
<td>3</td>
<td>Phoenix validates the restore destination (Original, alternate, or availability group) and the location on the restore destination.</td>
</tr>
</tbody>
</table>
Step | Operation
--- | ---
| | **Note**: Ensure that the location is not a drive. For example, C:\. To restore a database, restore the database files to a sub-folder. For example, C:\DBFiles.

| 4 | Phoenix validates if the SQL Server instance is available for restore. If the restore job replaces the original database, Phoenix also checks if the database is available for restore.

Phoenix checks the free space available for restore and starts the restore job by identifying filesets for restore. Phoenix sequentially downloads filesets to the restore destination. Within a fileset (which might contain data belonging to more than one database), Phoenix simultaneously downloads files that belong to different databases. After the download of a fileset completes, Phoenix downloads the next fileset (which might also contain data that belongs to one or more databases) to the restore destination.

| 6 | Phoenix uses the following syntax for the restore: `<destination path><snapshot><Request ID><Fileset><Actual file>`. The `<Request ID>` folder uniquely identifies each restore job.

The Phoenix agent picks up the transaction logs from the restore location, and:

- Picks the user-specified date and time (or the selected marked transaction)
- Checks which mode is enabled for restore:
  - If **Recovery mode** is enabled, Phoenix restores the database and leaves the database in the operational state (you cannot add more backup sets to it).
  - If **No recovery mode** is enabled, Phoenix restores the database and leaves the database in a pending state that cannot be accessed.
  - If **Standby mode** is enabled, it identifies the transaction closest to the date and time specified. Phoenix restores the database changes up to the identified transaction using the transaction log. In addition, it downloads the log files containing the transactions after the point-in-time restore.

If the original database is getting replaced, the Phoenix agent terminates all connections to the database and then restores it using the downloaded transaction logs.

If the database is restored to an availability group, the Phoenix agent either restores the database to the specified shared network location and syncs the database to all the secondary nodes or uses the automatic seeding if available.

After the restore completes:

- If **Recovery mode** is selected, the database appears as active for the instance that you specified.
- If **No recovery mode** is selected, the database appears as **Restoring** for the instance that you specified.
- If **Standby mode to allow restore of additional transaction logs** is selected, the database appears as Standby/Read-Only for the instance that you specified.
- If **Restore to Availability Group** is selected, the database appears as synchronized on all the nodes of the AG.
Restoring the database to a marked transaction

In addition, you can restore your database if you mark a transaction in SQL server. You can use the marked transaction to restore your database. Phoenix lets you choose a marked transaction to perform a point in time restore. The marked transaction restore, like the point in time restore using log backup, lets you restore your databases to a consistent state until a particular transaction occurs that updates your databases. Marked transaction restores use log backups that are incrementally triggered after a full backup job has completed, and a restore point exists.

Restore to original server in case of an AG, restores the database to the instance of the primary server.

MS-SQL server does not replicate transaction marks related to the transaction logs on the secondary nodes for databases in AG. If you try to restore a transaction mark for an AG backup set from Phoenix, you might or might not see transaction marks that were created on the MS-SQL server depending on whether the log backup happened from the primary or secondary node.

To restore your database using a marked transaction

1. Log in to the Phoenix Management Console.
2. On the menu bar, click Protect > MS-SQL Servers.
3. On the SQL Backup Sets tab, click the resource name that hosts the backup set that you want to restore.
4. On the server page, click Restore > Transaction Mark Restore on the backup set that you want to restore.
5. On the Restore to a Transaction Mark window, you can choose to restore original server instance or an alternate server instance.
   - If you choose original instance, provide the full path of the folder where you want to restore your database and click Next.
   - If you choose an alternate instance, provide the server, instance, and full folder path to restore your database and click Next.
6. The Restore to a Transaction Mark window lets you choose the marked transactions you can use to restore your database to a specific transaction. In addition, the restore options Stop Before Mark and Stop At Mark let you include the marked transaction in your restore job or exclude it.
   - After you select the transaction and the restore option, click Next.

In the next step, select the databases that use the marked transaction. Select the databases you want to restore, and click Done.

Transaction mark restore can fail with SQL8 error if the transaction marks were recorded before logs were backed up.