

## Chapter 13: Time Travel, Fail-Safe & Zero-Copy Cloning

So far, we have seen many features that make Snowflake an attractive service, but the features that we will see in this chapter make Snowflake unique! Let's take a look at Snowflake storage feature (Time Travel, Fail-Safe & Zero Copy Cloning), and you'll understand why Snowflake is getting more and more popular lately!

1. [Introduction to Snowflake Storage Features](#)
2. [Time Travel](#)
3. [Restore Objects using Time Travel](#)
4. [Querying over Historical Data using Time Travel](#)
5. [Fail-Safe](#)
6. [Zero-Copy Cloning](#)
7. [Typical Exam Questions](#)

### INTRODUCTION TO SNOWFLAKE STORAGE FEATURES

Have you ever seen the meme (I hope it hasn't happened to you) where you start working at 9 a.m. on a Monday, and you delete a table in a production environment? Well, this will no longer be a problem thanks to Snowflake!

In the following diagram, we can see different ways to recover data in Snowflake, first of all by ourselves using the Time Travel functionality, and after that time by Snowflake support using the Fail-Safe retention period. Let's study these concepts!

## Continuous Data Protection Lifecycle

**Standard operations allowed**  
Queries, DDL, DML, etc.

**Time Travel SQL extensions allowed**  
SELECT ... AT|BEFORE ...  
CLONE ... AT|BEFORE ...  
UNDROP ...

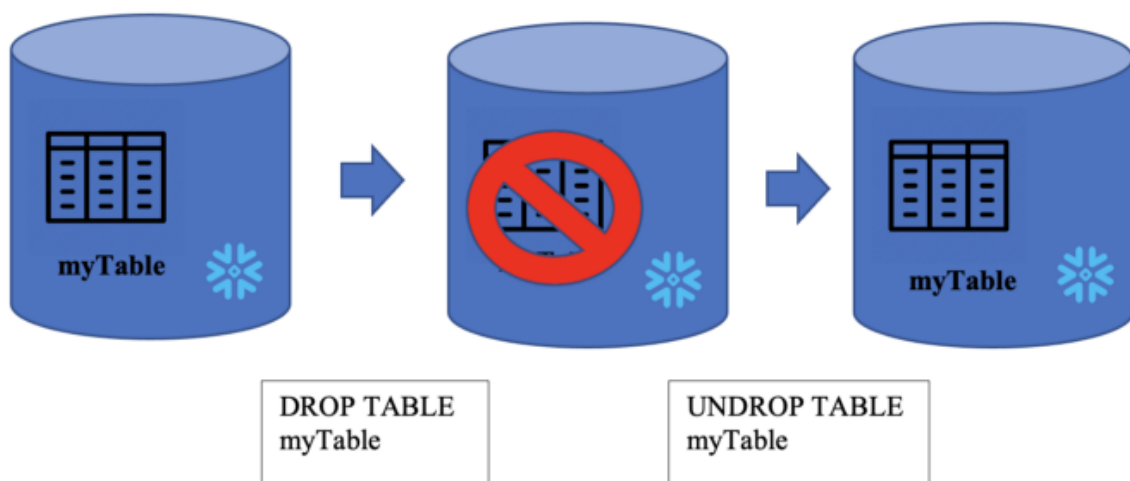
**No user operations allowed**  
(data recoverable only by  
Snowflake)



Recovering data using Snowflake (via [docs.snowflake.com](https://docs.snowflake.com)).

## TIME TRAVEL

Time travel enables accessing historical data (i.e., data that has been changed or deleted) at any point within a defined period. If we drop a table, we can restore it with time travel. You can use it with Databases, Schemas & Tables.



Time Travel in Snowflake

### Use cases of Time Travel:

- Restoring data-related objects that we might have accidentally or intentionally deleted.
- Duplicating and backing up data from key points in the past.
- Query data from the past.

No tasks are required to enable Time Travel. By default, it's enabled with a 1-day retention period. However, we can increase to 90 days if we have (at least) the Snowflake Enterprise Edition. It requires additional storage, which will be reflected in your monthly storage charges.

### RESTORE OBJECTS USING TIME TRAVEL

To restore objects, we use the command "UNDROP". As we mentioned before, you can use it with Databases, Schemas, or Tables. If we try to restore an object with a name that already exists, Snowflake will give an error.

`DROP TABLE mytable;` -> We accidentally drop a table

`UNDROP TABLE mytable;` -> We restore it using Time Travel

In this example, we try to `UNDROP` an object with the same name, producing an error.

```
DROP TABLE mytable;
```

```
CREATE TABLE mytable (c1 varchar);
```

*UNDROP TABLE mytable; -> It will trigger an error*

## QUERYING OVER HISTORICAL DATA

*We can also query over historical data. Imagine that a table has changed a lot during the last week, and we want to query how it was seven days ago. We can also do that thanks to Time Travel. There are different ways to do it:*

- *By timestamp*

```
SELECT * FROM my_table AT(timestamp => 'Mon, 01 May 2021  
08:00:00 -0700'::timestamp_tz);
```

- *By offset → In this example, we select the historical data from a table as of 15 minutes ago:*

```
SELECT * FROM my_table AT(offset => -60*15);
```

- *By query statement ID → We can see the Query Statement ID in the history of queries:*

```
SELECT * FROM my_table BEFORE (STATEMENT =>
'8e5d0ca9-005e-44e6-b858-a8f5b37c5726');
```

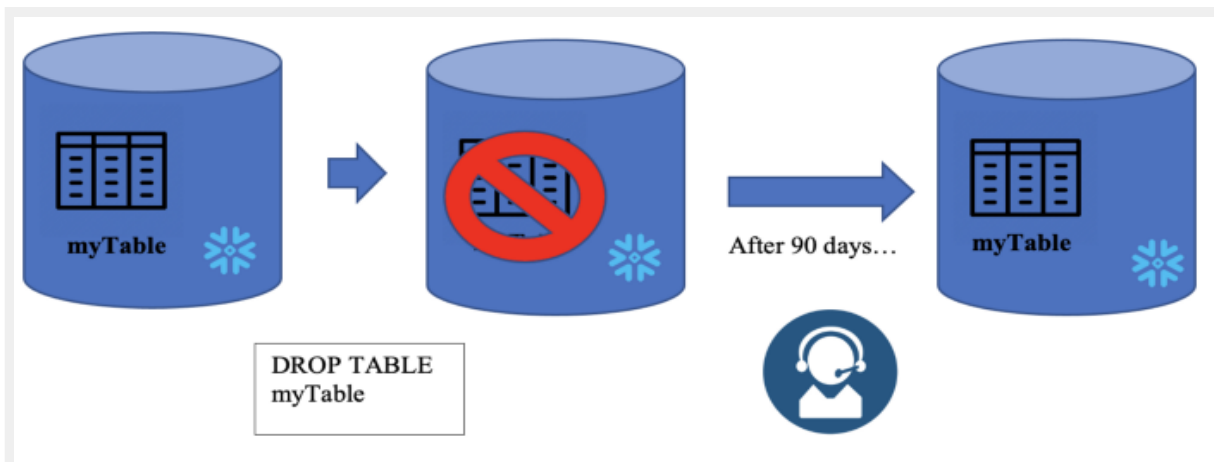
The screenshot shows the Snowflake History page. The top navigation bar includes icons for Databases, Shares, Data Marketplace, Warehouses, Worksheets, History (selected), Preview App, and Partner. Below the navigation bar, the 'History' tab is active. A search bar contains 'GONZALO FERNANDEZ'. Below the search bar, there are checkboxes for 'Include client-generated statements' and 'Include queries executed by user tasks'. A table displays query history with columns: Status, Query ID, and SQL Text. The first row shows a green checkmark in the Status column, the Query ID '020e4e3f-3201-de43-0560-32fd00b1355e' in the Query ID column, and 'select top 1000 \* fro...' in the SQL Text column. The Query ID cell is highlighted with a red rectangle.

Status	Query ID	SQL Text
✓	020e4e3f-3201-de43-0560-32fd00b1355e	select top 1000 * fro...

How to find the Snowflake Query Statement ID in the Snowflake UI.

## FAIL-SAFE

Fail-safe ensures historical data is protected in the event of a system failure or other catastrophic event, e.g., a hardware failure or security breach. It provides a (NON-CONFIGURABLE) 7-day period during which Snowflake support may recover historical data. This is important; you cannot recover this data by yourself, you have to ask the Snowflake support. This period starts immediately after the Time Travel retention period ends, and it also requires additional storage as Time Travel.



### Fail-Safe in Snowflake

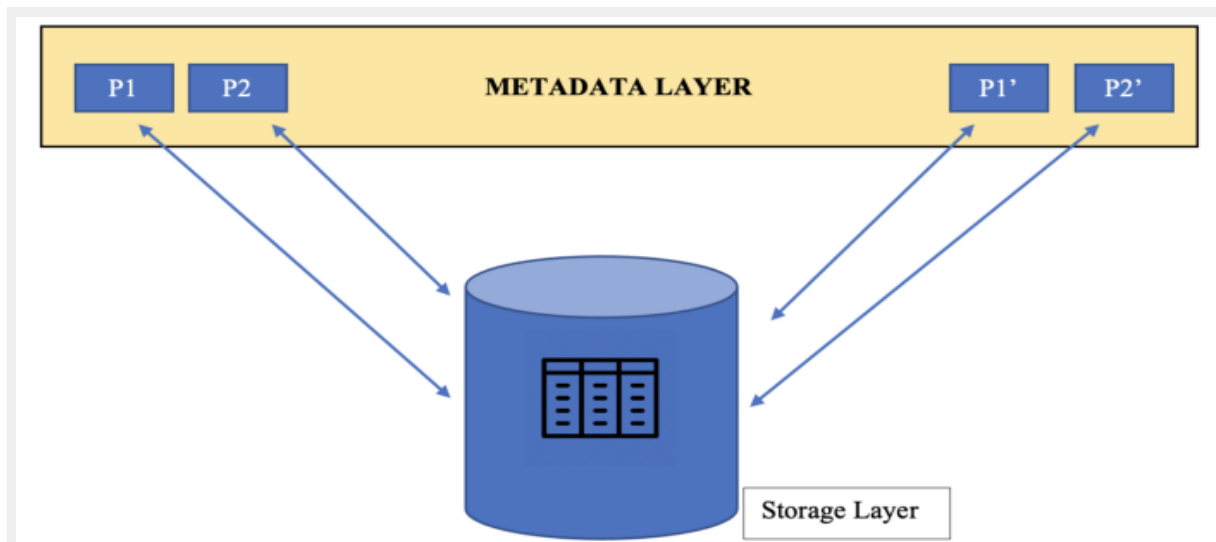
Imagine that we have 20 days Time Travel period. We would have all our data protected for 20 (Time Travel) + 7 (Fail Safe) days. If the Time Travel period is 90 days, Snowflake will protect the data for 97 days.

Only permanent tables have Fail-Safe period, as we saw in this table.

## ZERO-COPY CLONING

Using Zero-Copy cloning, you can create a snapshot of any table, schema, or Database. The cloned object is independent and can be modified without modifying the original one. Very good for development jobs.

Zero-Copy cloning does NOT duplicate data; it duplicates the metadata of the micro-partitions. For this reason, Zero-Copy cloning is FREE, as it doesn't consume storage. When you modify some data of the clone, it will consume storage because Snowflake has to recreate the micro-partitions, which will cost money. As an example, if you are only going to perform read queries, there are no additional storage costs.



### Zero Copy Cloning in Snowflake

Some considerations that might appear in the exam:

- Privileges are not cloned.
- Named Internal Stages are not cloned, but external ones are cloned. Table Stages are also cloned.
- Pipes that reference the internal stages are not cloned. If they reference an external stage, they are cloned.
- Data History is not cloned.

Depending on the object you are going to clone, you need some privileges. They are shown in the following table:

	Table	Pipe	Stream	Task	Other objects
Needed Privilege	SELECT	OWNERSHIP	OWNERSHIP	OWNERSHIP	USAGE

Privileges required for Zero-Copy Clone.

## TYPICAL EXAM QUESTIONS

1. What type of data incur in Snowflake Storage costs?

1. Data stored in permanent tables
2. Fail-Safe data
3. Zero-copy cloning data
4. Time travel data
5. Data stored in cloud storage like AWS
6. Data stored in temporary tables

**Solution:** 1, 2, 4, 6. Remember that for the zero-copy clone, it clones the metadata from the micro-partitions, not the real data. That's why it doesn't incur storage costs. Temporary tables also have a storage cost, as we saw in its chapter.

	Time Travel	Fail Safe	Cloning	Create views
Permanent	0-90 days.	Si	Permanent Transient Temporary	Yes
Transient	0-1 days.	No	Transient Temporary	Yes
Temporary	0-1 days.	No	Transient Temporary	Yes
External	No	No	No	Yes

Difference between Snowflake tables.

2. What should be the first option to restore data into a table?



1. Time-Travel
2. Fail-Safe
3. Zero-Copy Cloning

*Solution: 1.*

*3. Which Snowflake edition allows only one day of Time Travel?*

1. Standard
2. Enterprise
3. Business Critical

*Solution: 1*

*4. Which Snowflake edition (and above) allows until 90 days of Time Travel?*

1. Standard
2. Enterprise
3. Business Critical

*Solution: 2*

*5. Do you need to contact Snowflake Support to retrieve data from Fail-Safe?*

1. True

2. False

*Solution: 1*

*6. Is Fail-Safe period for temporary and transient tables 0?*

1. True

2. False

*Solution: 1. Their Time Travel period is either 0 or 1 day, but they don't have Fail-Safe retention period.*

*7. Select the statements that are true about Fail-Safe:*

1. There is no difference between Time-Travel and Fail-Safe

2. The data retention period for a permanent table with 30 days of Time-Travel is 37 days

3. Fail-Safe provides a non-configurable 7-day period

4. Fail-Safe provides a non-configurable 90-day period

5. Fail-Safe ensures that historical data is protected in the event of a system failure or other catastrophic events

*Solution: 2, 3, 5. Regarding the second option, we have 30 days of Time-Travel, apart from 7 days of Fail-Safe. 37 days in total.*

*8. A point-in-time snapshot of data that can be updated by users is called...*

1. Time-Travel
2. Fail-Safe
3. Zero-Copy Cloning

*Solution: 3.*

*9. Which service does Snowflake use to provide the Zero-Copy cloning functionality?*

1. SSD Cache of the Virtual Warehouses
2. Cache
3. Metadata from the service layer

*Solution: 3*

*10. What would happen if we executed the following command?*

```
CREATE OR REPLACE TABLE newTable CLONE table1;
```

1. "newTable" is created with all the data from "table1"
2. "newTable" is created, and Snowflake internally executes a batch job to copy all the data from "table1"
3. "newTable" is created, and Snowflake internally executes a pipe to copy all the data from "table1"
4. Snowflake creates a new entry in the metadata store to keep track of the new clone. The existing micro-partitions of "table1" are mapped to the new table.

*Solution: 4. This is how zero-copy cloning works.*

*Thanks for Reading!*