# Recovery and Backup Guide

## Recovery for Tables in Databricks

This section summarizes different types of tables in Databricks, where they are stored, how they are managed, and their recovery options in case of mistakes or disasters.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Type | Definition | Storage Location | Managing Catalog | Recovery for Mistake | Recovery for Disaster |
| Delta Live Tables | Not a physical table. Used as a framework for building data pipelines. | n/a | n/a | n/a | n/a |
| Managed Tables | Default Databricks tables. Managed entirely in Delta Lake and Unity Catalog. | AWS (internal) | Unity Catalog | Undo within 7 days using time travel or RESTORE | Backup/Restore or Deep Clone to external storage |
| External Tables | Unmanaged tables. Data is stored outside Databricks (e.g., S3). | External Storage | External System or Unity Catalog | Based on external system retention | Depends on external backup/restore strategy |
| Foreign Tables | Federated tables from external systems, managed by foreign catalogs. | External System | External System or Unity Catalog | Based on external system | Depends on source system strategy |
| Delta Tables | Primary table type in Databricks. Supports multiple features and customizations. | AWS | Unity Catalog | Time travel functionality for rollback | Backup/Restore or Deep Clone to external storage |
| Streaming Tables | Append-only Delta tables designed for high-throughput pipelines. | AWS | Unity Catalog or DLT | Time travel functionality | Backup/Restore or Deep Clone to external storage |
| Feature Tables | ML-specific Delta tables with primary keys for model tracking and governance. | AWS | Unity Catalog | Time travel functionality | Backup/Restore or Deep Clone to external storage |
| Online Tables | Read-only Delta tables accessed via API or Explorer. | AWS | Unity Catalog | Time travel functionality | Backup/Restore or Deep Clone to external storage |

Helpful Links:

• Delta Tables: https://docs.databricks.com/en/delta/index.html

• AWS Backup: https://docs.aws.amazon.com/aws-backup/latest/devguide/whatisbackup.html

## AWS S3 Backup and Restore

Amazon S3 is a scalable storage service. AWS provides several methods for backing up and restoring S3 buckets.

### Why Backup S3 Buckets?

- Ensure data protection from accidental deletion or corruption.

- Comply with regulatory data retention requirements.

### Backup Methods

1. Versioning: Retains multiple versions of a file in the same bucket.

2. Cross-Region Replication (CRR): Replicates S3 bucket to another AWS region.

3. Lifecycle Policies: Manages file transitions and deletions automatically.

4. AWS Backup: Centralized and automated backup management.

5. Third-Party Tools: Tools like CloudBerry, Druva offer advanced features.

### Steps to Backup S3 Buckets

1. Enable Versioning via S3 bucket settings.

2. Configure CRR in the S3 Management Console.

3. Set Lifecycle Policies for archival/deletion.

4. Use AWS Backup service to manage backups.

### Restore from AWS Backup

1. Access AWS Backup and find the backup vault.

2. Select a recovery point.

3. Restore to a new or existing bucket.

4. Monitor the restore job and verify data integrity.

### Tips

- Assign appropriate IAM roles for restore operations.

- Enable notifications for job statuses.

- Periodically test restore to ensure data recoverability.

Reference Links:

• AWS Backup Overview: https://docs.aws.amazon.com/aws-backup/latest/devguide/whatisbackup.html

• Databricks Delta Time Travel: https://docs.databricks.com/en/delta/history.html#time-travel