Point-in-Time Recovery in YugabyteDB

Sergei Politov
Friday, Apr/1/2022

YFTT
YUGABYTEDB
FRIDAY
TECH TALKS

yugabyteDB

Point-in-Time Recovery in YugabyteDB

Purpose of Point-in-Time Recovery

Point-in-Time Recovery provides ability to restore database state to any point in the specified time window.

It can be used to recover the database in case of unintended changes:

- Data modification (insert/update/delete)
- Table or index create/delete
- Schema changes
- Etc.

Point-in-Time Recovery in YugabyteDB

Options

Point-in-Time Recovery is managed using so called snapshot schedules. Each snapshot schedule has the following options:

- Interval - How frequently snapshot should be created. We cannot garbage collect MVCC history after the last snapshot; i.e. cannot completely cleanup deleted or updated rows. So it is not recommended to put it too high. I would suggest 60 minutes for interval, but it depends on workload.
- Retention - How long we should keep snapshots and provide ability to recover, i.e. we would not be able to recover past this period.
- Database - snapshot schedule will operate on this database.

Point-in-Time Recovery in YugabyteDB

Under the Hood

- ○ Snapshots are used for data recovery. Snapshot is implemented using hard links and could be created in milliseconds. Providing minimal overhead in disk usage.
- ○ Because of Multiversion concurrency control (MVCC) in YugabyteDB, it is possible to read snapshot data state at any specified time.
- ○ In order to restore deleted table, table is not actually deleted, but hidden from client. And cleaned up after retention period.
- ○ To restore postgres system catalog, we calculate delta between current catalog version and version at the restoration time. Then apply it to current catalog.

Point-in-Time Recovery Demo

Thank You

Join us on Slack: yugabyte.com/slack (#yftt channel)

Star us on Github: github.com/yugabyte/yugabyte-db

YFTT
YUGABYTEDB
FRIDAY
TECH TALKS

yugabyteDB