**Hot Backup in PostgreSQL**

Yes, **you can take a backup of a running PostgreSQL database**, and PostgreSQL fully supports this.

There are **two main types** of backups you can use on a live database:

**1. Logical Backup (e.g., pg\_dump)**

* Works on a **running database**
* Dumps SQL statements or custom formats
* Can back up full DB or specific tables/schemas
* **No need to stop or lock the database**

**Example:**

pg\_dump -U postgres -d mydb -F c -f /tmp/mydb.backup

* -F c: custom format
* -f: output file

You can continue to **read/write** during this.

**How it Works**

* pg\_dump takes a **snapshot** of the database schema and data at a **specific point in time**.
* It uses **MVCC (Multi-Version Concurrency Control)** to **read consistent data** even while writes are happening.
* It queries the database like a normal client, issuing SELECT statements.

**Consistency Guarantee**

* It uses a **single transaction** for the entire dump to ensure consistency.
* Since it's read-only, it **does not block writes**, and writers don't block it.

**2. Physical Backup (e.g., pg\_basebackup)**

* Works **on running databases**
* Backs up the entire **data directory**
* Requires **WAL archiving** or replication to be enabled
* Often used for **PITR (Point-in-Time Recovery)** or replica setup

**Example:**

pg\_basebackup -U replication\_user -D /var/lib/postgresql/backup/ -Ft -z -P

* -Ft: tar format
* -z: compress
* -P: progress
* Requires replication\_user with REPLICATION privileges

**How it Works**

* pg\_basebackup copies the actual **PostgreSQL data directory**, including WAL (Write-Ahead Logs).
* It starts the backup by issuing pg\_start\_backup() internally (unless in a replica mode).
* During the backup, **WAL files are collected**, which record all changes.
* After the data files are copied, it finalizes the backup with pg\_stop\_backup().

**Consistency Guarantee**

* WALs ensure that the backup is **crash-consistent** and restorable to a valid state.
* When restoring, PostgreSQL replays WALs to bring the backup to a consistent point.

**Key Notes**

| **Type** | **Hot Backup?** | **Supports Restore to Another Version?** | **Suitable For** |
| --- | --- | --- | --- |
| pg\_dump | Yes | (cross-version) | Logical export/import |
| pg\_basebackup | Yes | (same version only) | Full physical copy / PITR |