LSN

The Log Sequence Number (LSN) in PostgreSQL is a critical part of its Write-Ahead Logging (WAL) mechanism, ensuring data integrity and consistency. Here’s a detailed explanation of how an LSN is created, how it tracks changes, and how it fits into the correct WAL segment and table.

LSN Creation:

1. Transaction Start:
— When a transaction starts, PostgreSQL assigns it a Transaction ID (XID).
— As the transaction proceeds and modifies data, these changes are recorded in the WAL buffer.

2. Assigning LSN:
— Each change (e.g., insertion, update, or deletion) generates a WAL record.
— When the WAL record is created, it is assigned a unique LSN.
— An LSN is a 64-bit number represented in hexadecimal, divided into two parts: log ID and offset (e.g., `0/169EC40`).

Writing to WAL Buffer and WAL Segment

1. WAL Buffer:
— The WAL buffer is an in-memory circular buffer that temporarily holds WAL records.
— When a change is made, the corresponding WAL record is written to the WAL buffer with its assigned LSN.

2. Flushing to WAL Segment:
— WAL records in the WAL buffer are periodically flushed to WAL segment files on disk.
— A WAL segment is a fixed-size file (usually 16MB) where WAL records are stored.
— The flushing occurs:
— When the WAL buffer is full.
— During a commit (depending on `synchronous_commit` setting).
— During checkpoints.

Determining the WAL Segment

- Log ID and Offset:
— The LSN's log ID part determines which WAL segment file

the record will go into.
— The offset part determines the position within that WAL segment.

- WAL Segment Naming:
— WAL segments are named based on the log ID and offset. For example, `000000010000000000000001` where the first part is the timeline, the second part is the log ID, and the third part is the segment ID.

LSN and Table Relationship

1. Table Page Changes:
— When a row in a table is modified, the corresponding page in the shared buffer is updated.
— The change is recorded in the WAL buffer with an LSN.
— This WAL record includes information about the table and the specific page that was modified.

2. Page Redo:
— During recovery, PostgreSQL uses the LSN to replay WAL records.
— Each WAL record includes enough information to identify the table and the page within the table that needs to be modified.
— The system applies the changes to the appropriate table pages to bring them to a consistent state.

Example Scenario

1. Transaction Insert:
— A transaction inserts a row into a table.
— PostgreSQL assigns an XID to the transaction.
— A WAL record for the insert is created and assigned an LSN (`0/169EC40`).

2. WAL Buffer:
— The WAL record is written to the WAL buffer.

3. Flushing to WAL Segment:
— The WAL buffer is periodically flushed to a WAL segment file named `000000010000000000000001`.
— The LSN (`0/169EC40`) determines its position within the segment.

4. Commit:
— The transaction commits, ensuring that the WAL records are flushed to disk (based on `synchronous_commit`).

Diagram of the Process

```

Transaction Start

+ — — — — — — — — — — — — +
| Assign XID (1001) |
+ — — — — — — — — — — — — +

Insert into Table
```

+ — — — — — — — — — — — + + — — — — — — — — — — — + + — — — — — — — — — — — — +
| Create WAL Record | → | Assign LSN (0/169EC40) | → | Write to WAL Buffer |
+ — — — — — — — — — — — + + — — — — — — — — — — — + + — — — — — — — — — — — — +

Periodic Flush to WAL Segment

+ — — — — — — — — — — — — — — — — -+
| WAL Segment: 000000010000000000000001 |
+ — — — — — — — — — — — — — — — — -+
| LSN (0/169EC40) | WAL Record (Insert) |
+ — — — — — — — — — — — — — — — — -+

Transaction Commit

+ — — — — — — — — — — — — — — — — -+
| Flush WAL Buffer to Disk |
+ — — — — — — — — — — — — — — — — -+

Recovery (if needed)

+ — — — — — — — — — — — — — — — — -+
| Read Last Checkpoint |
+ — — — — — — — — — — — — — — — — -+
| Replay WAL Records from LSN |
+ — — — — — — — — — — — — — — — — -+
| Apply Changes to Table Pages|

```
+———————————————-+
```

Summary:

- LSN Creation: An LSN is assigned to each WAL record when a change occurs.
- WAL Buffer: WAL records are written to the WAL buffer with their LSNs.
- WAL Segment: WAL records are periodically flushed from the WAL buffer to WAL segments on disk.
- Table Pages: WAL records include information about the specific table and page modified, allowing PostgreSQL to replay these changes during recovery.

This mechanism ensures that all changes are logged in the correct order, making recovery possible even after an unexpected shutdown.