


# System Catalog & Fast DDL

---

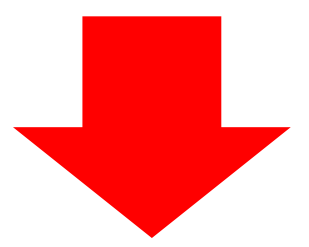
15-721 Final Presentation

Mengran Wang, Ang Li, Yixin Luo

# System Catalog Object

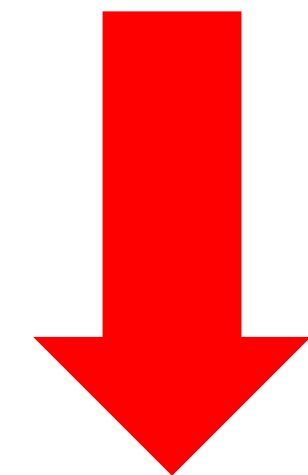
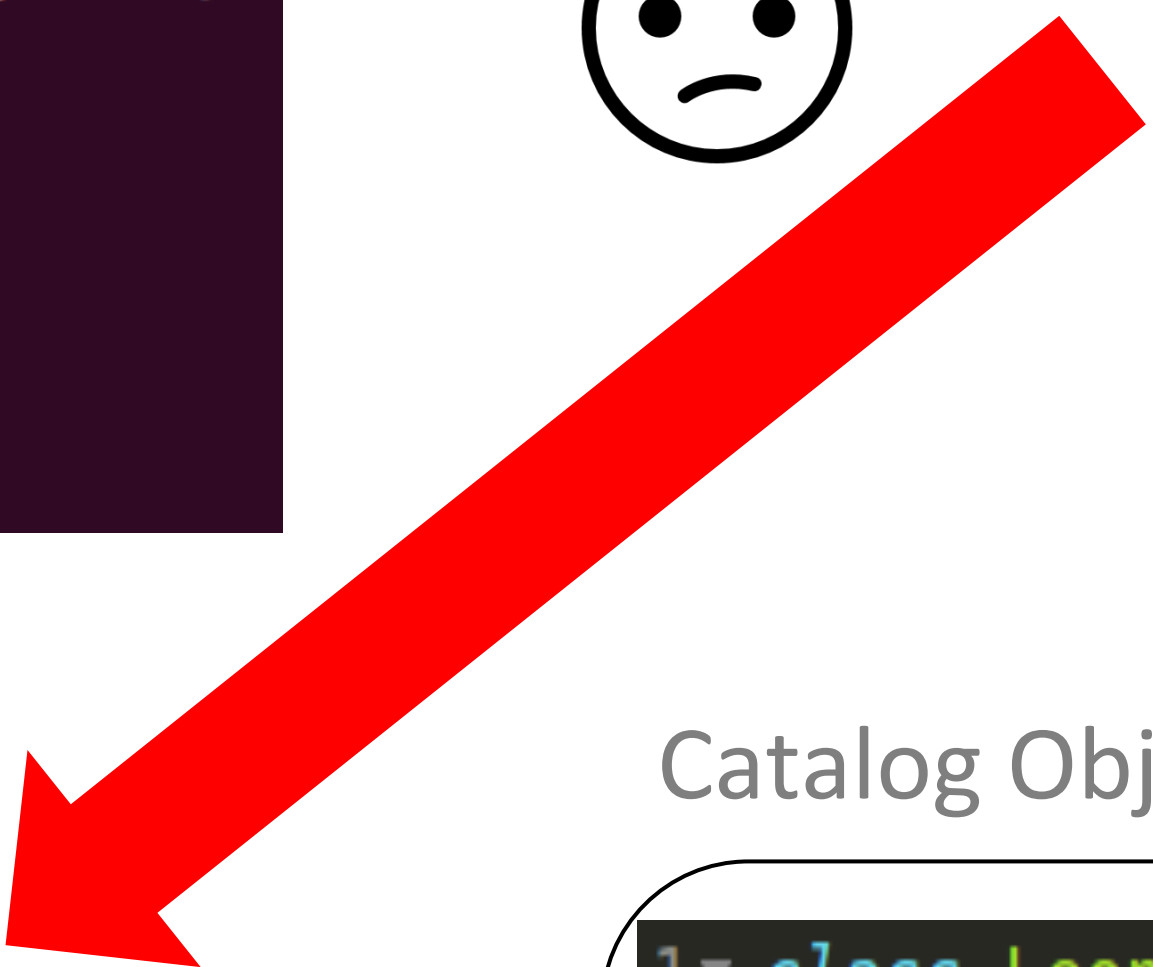


```
pg_catalog
postgres=# select * from pg_leon;
table_oid | leons
-----+-----
      100 |      1
      101 |      2
(2 rows)
```



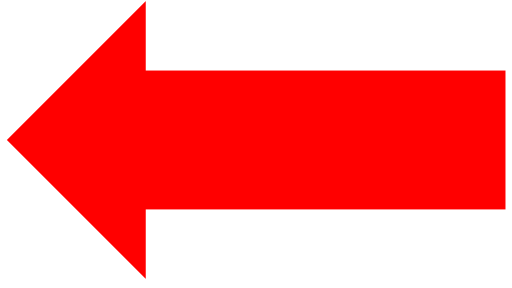
Catalog Table "pg\_leon"

table_oid	leons
100	1
101	2



Catalog Object

```
1 class LeonCatalog {
2   // Write API
3   bool InsertLeon(oid_t table_oid, int leons);
4   bool DeleteLeon(oid_t table_oid);
5   // Read API
6   int GetLeons(oid_t table_oid);
7 };
```



# System Catalog Demo

---

# DDL Operations Are Slow!

- ▶ Copying: DDL operations disrupt production systems and reduce throughput.
- ▶ Blocking: DDL operations block concurrent transactions on the same table.
- ▶ Some schema change can take hours!

# Naive ALTER TABLE with 2PL (Postgres)

- ▶ Table-level explicit locking
- ▶ Use hierarchical 2PL with deadlock detection
- ▶ Can be examined through *pg\_locks* system view

# Naive ALTER TABLE with 2PL (Postgres)

Access Exclusive Lock



Table Foo (old schema)

A	B	C
x	y	z
l	m	n
...	...	...

Translate & Copy

Table Foo (new schema)

A	B	C	D
x	y	z	0

# Lazy & Online ALTER TABLE

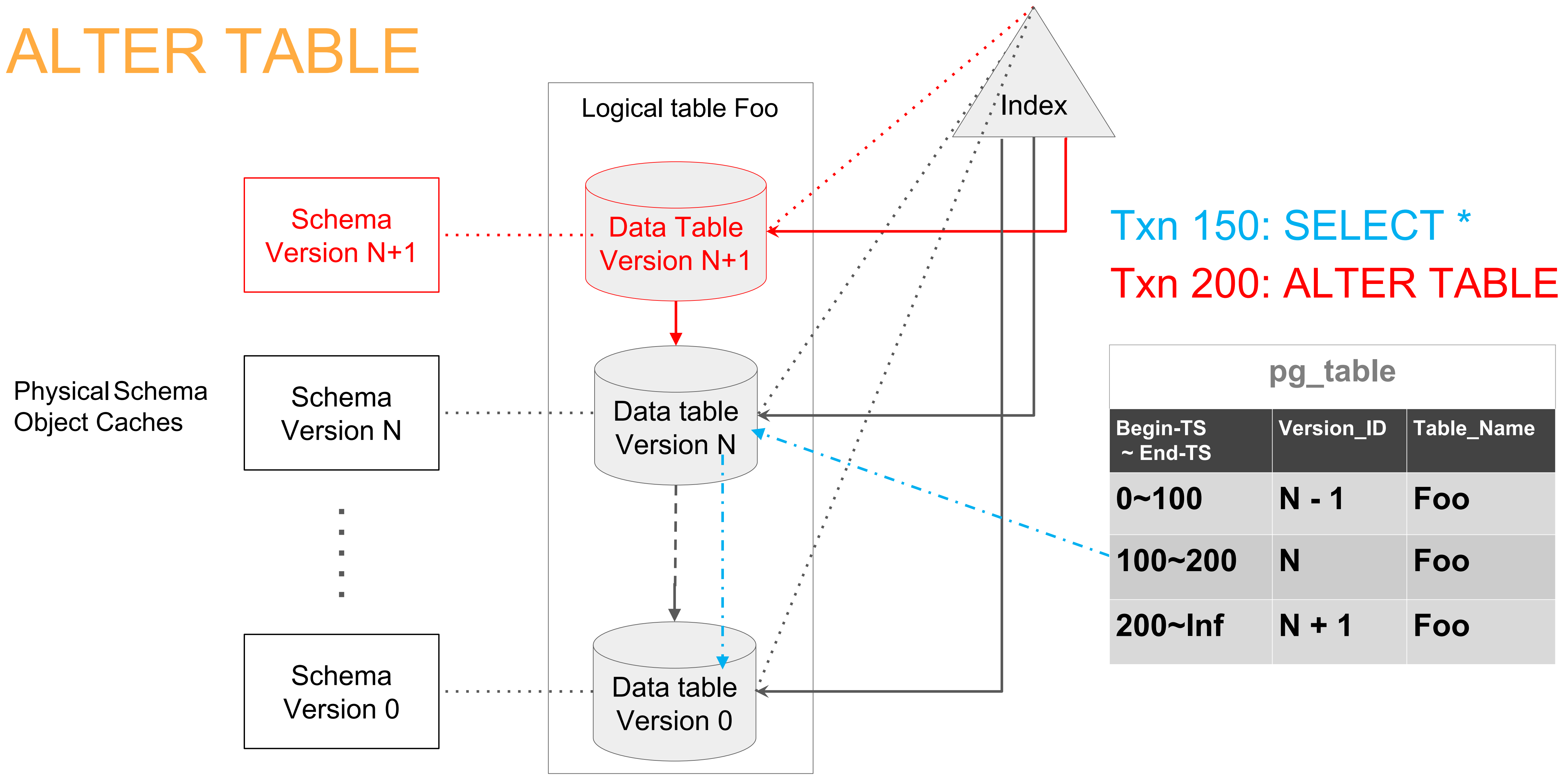
- ▶ How to achieve **lazy**?

Keep **old data** in place, but interpret with **new schema** (avoid **copying** data)

- ▶ How to achieve **online**?

Make schema **multi-versioned** (avoid **blocking** on concurrent transactions)

# ALTER TABLE

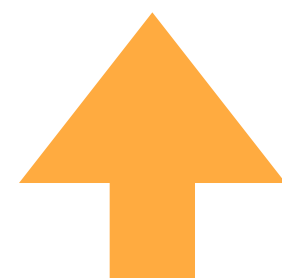




# Schema Mapping

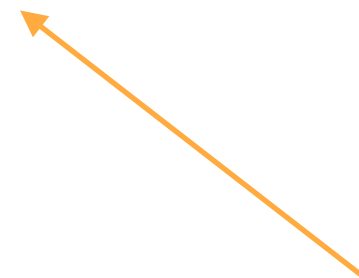
- ▶ Use **unique Column\_id** per logical table for schema mapping

Schema Version y



Schema Version x

Column 1 | Column 3



Column 1 | Column 2 | Column 3

Drop Column

Column 1 | Column 2 | Column 3



Column 1 | Column 2

Default Value

Add Column

# Tuple Migration

- ▶ When to update old data tuples with new schema?
  - ▶ On tuple access
  - ▶ On tuple update
  - ▶ Use a background worker

# Future Work

- ▶ Online ALTER TABLE working in progress
- ▶ ALTER TABLE benchmarking
- ▶ Support transactional DDL