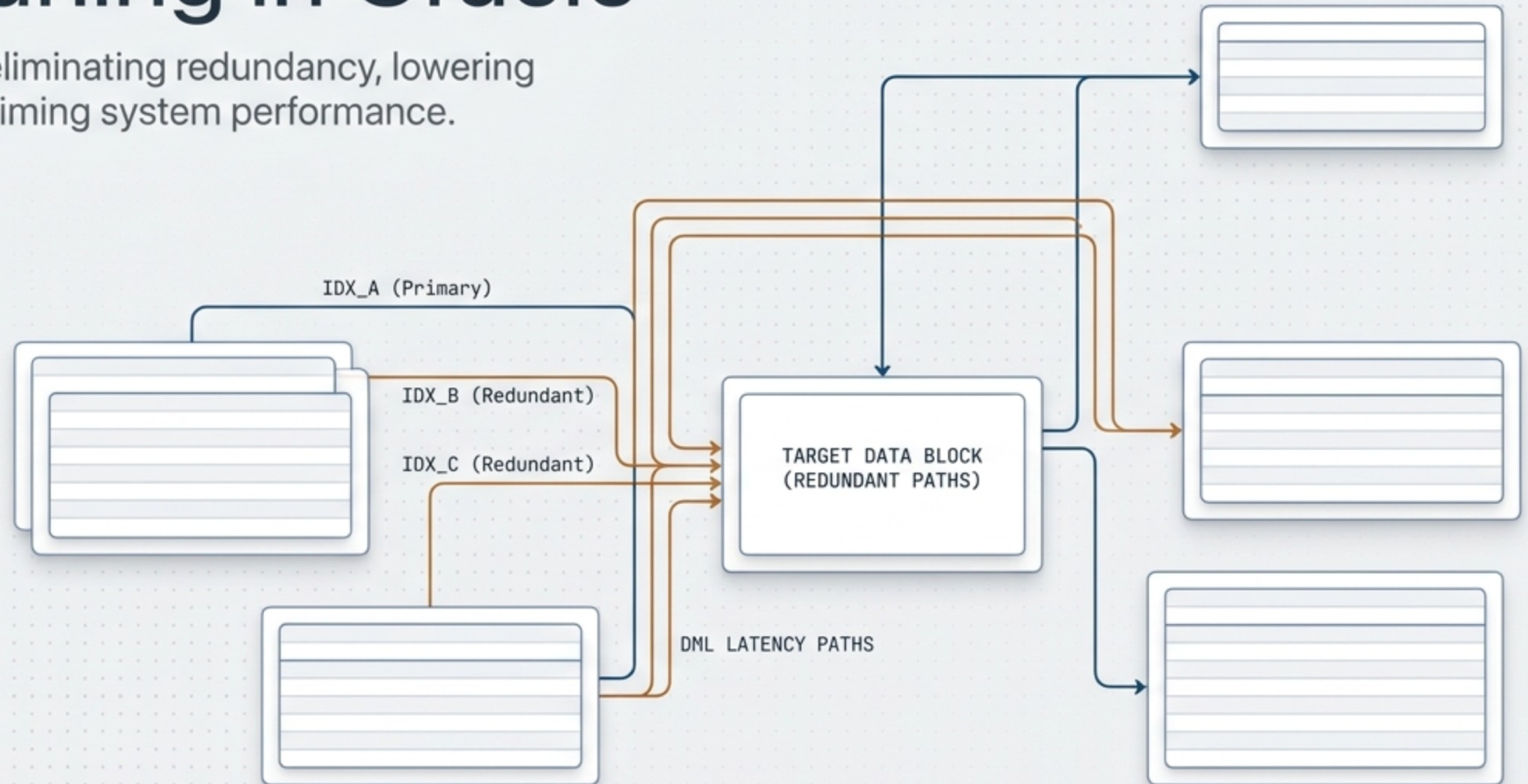
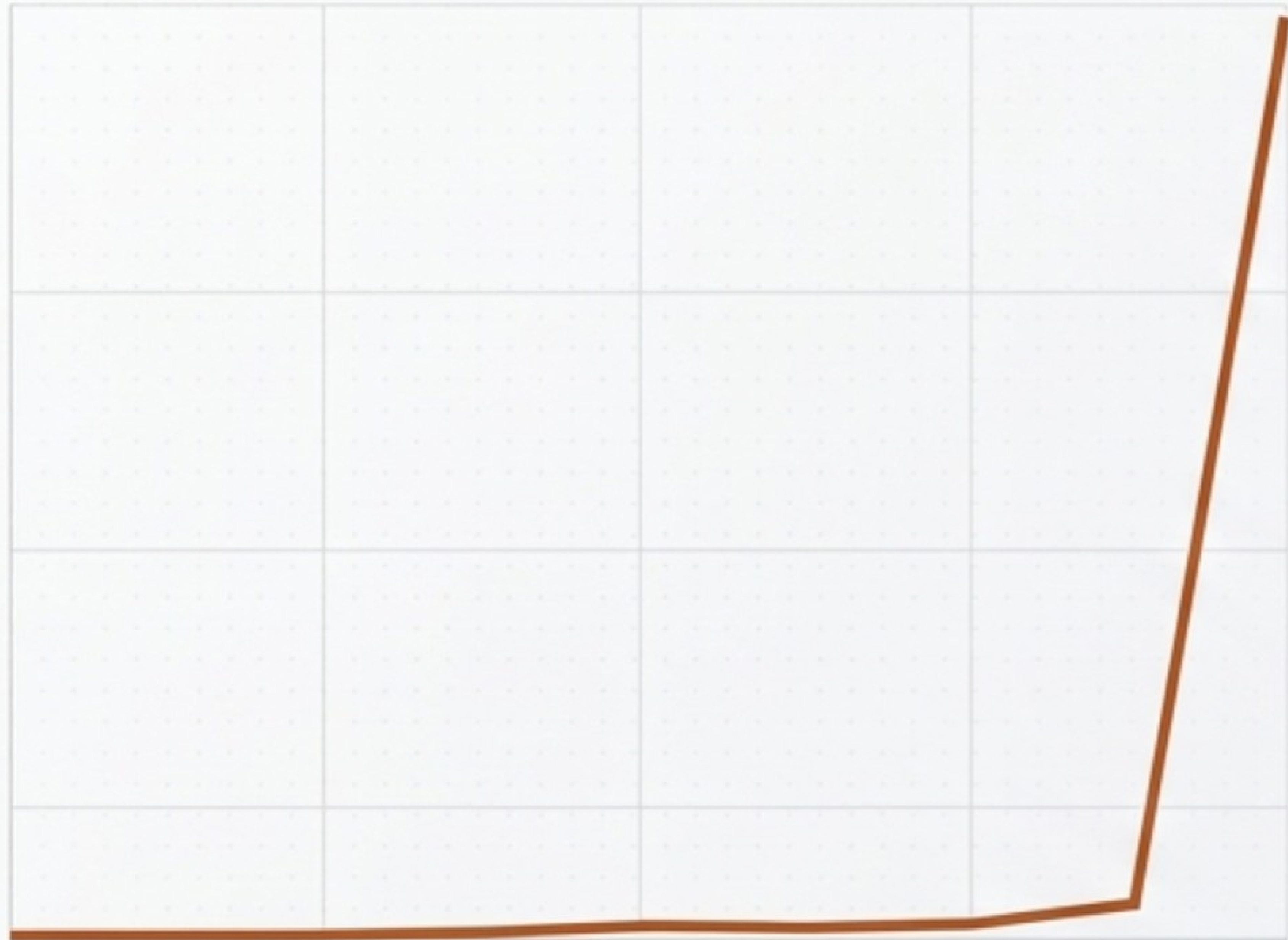


The Hidden Tax: Forensic Index Pruning in Oracle

A diagnostic guide to eliminating redundancy, lowering DML latency, and reclaiming system performance.



Redo Log Generation



The “Just-In-Case” Fallacy

“Every ‘just in case’ index adds a millisecond tax to every single insert.”

— DBA Forensics Podcast

Developers often build indexes defensively to speed up hypothetical reads. But this creates a hidden write tax. When multiple indexes map the same paths, they trigger massive, unnecessary spikes in redo log generation.

The 4 Pillars of System Degradation



DML Overhead

Every INSERT, UPDATE, and DELETE must maintain every index. Tripling an index triples the write work per row.



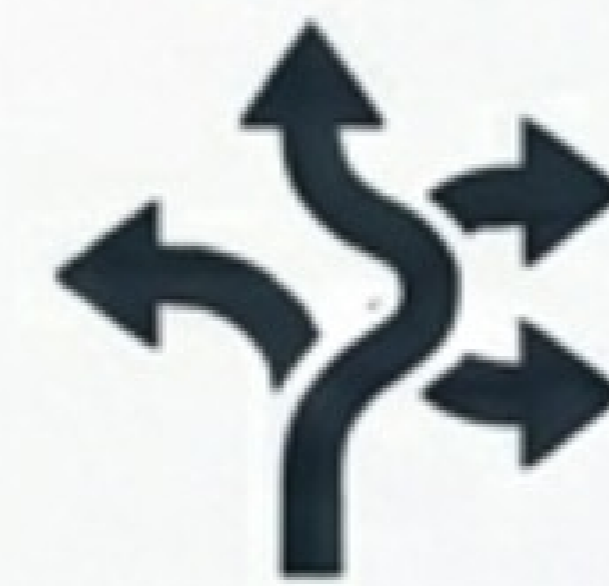
Storage Waste

Redundant indexes consume massive tablespace segments and artificially inflate backup sizes.



Buffer Cache Inefficiency

Duplicates compete for SGA (System Global Area) space, pushing out useful data and driving up physical I/O.

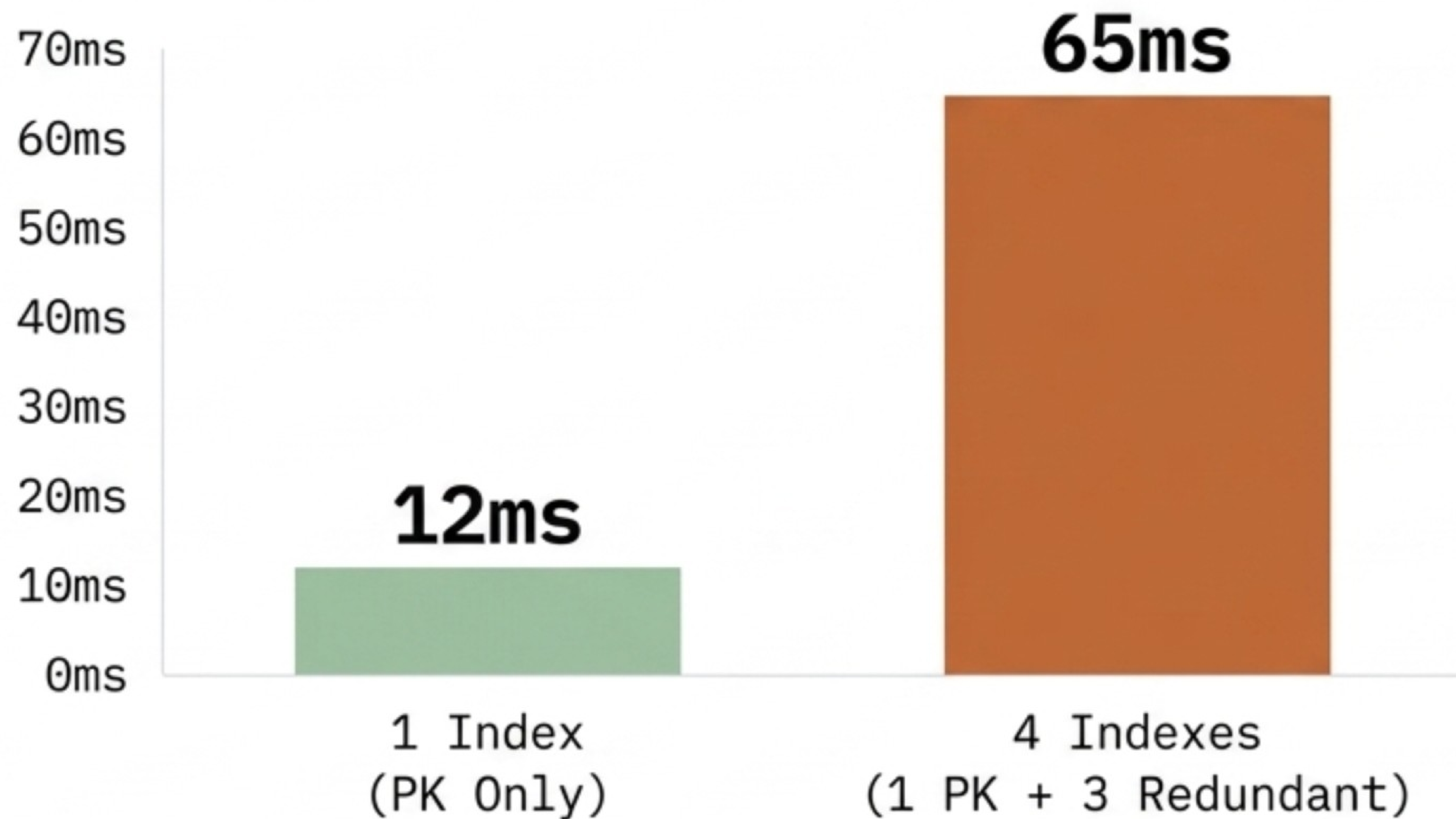


Optimizer Confusion

Multiple similar access paths can confuse the Cost-Based Optimizer (CBO), leading to sub-optimal execution plans.

Telemetry Proof: The Real-Time Cost

Simulated DML Performance Degradation



Simulation: 100,000 Row Insert. Adding 4 overlapping indexes multiplied write latency by 5x.

Diagnostic Matrix: The 3 Faces of Redundancy

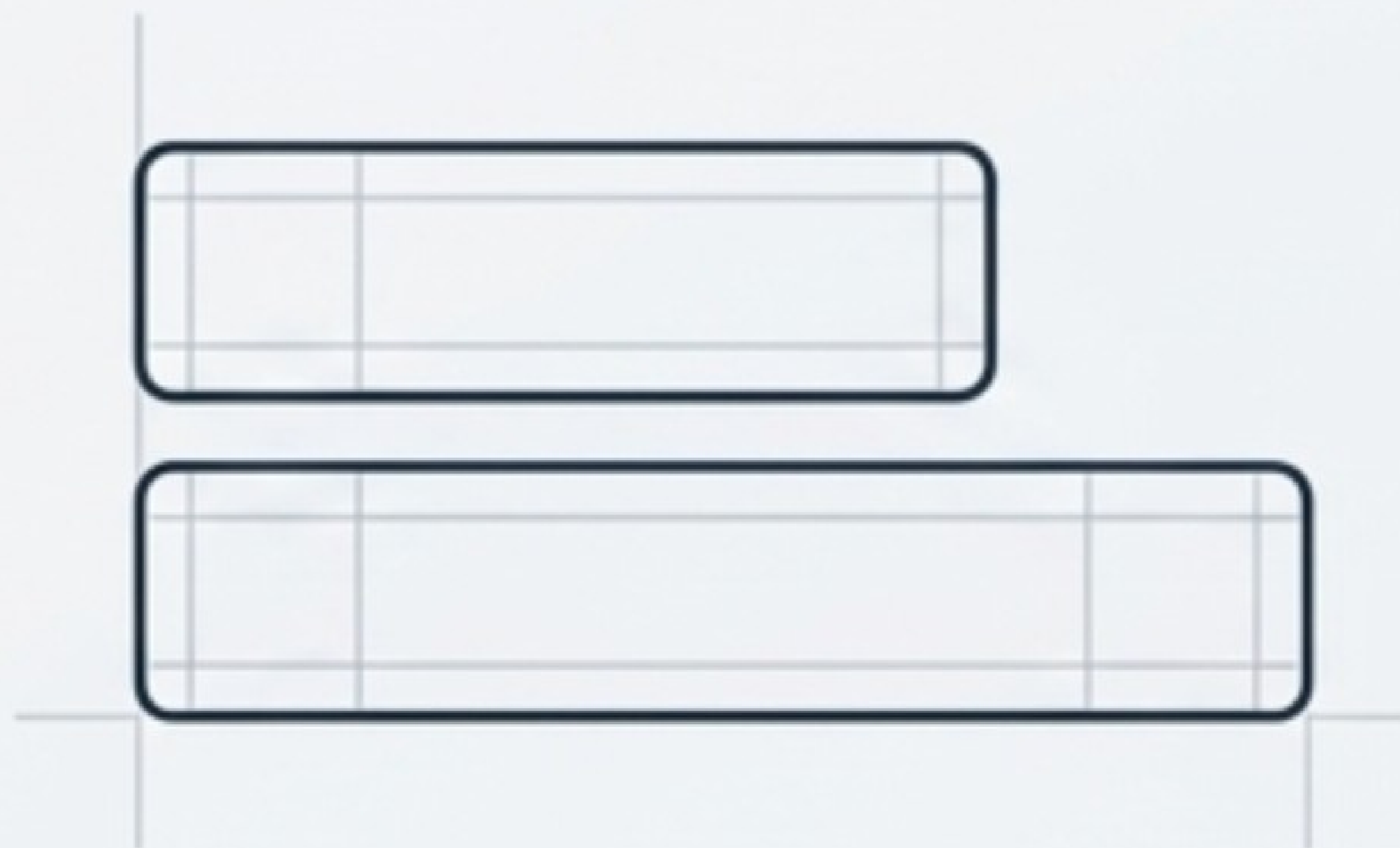
Exact Duplicates



Definition: Two indexes with the exact same columns in the exact same order.

Remediation: Immediate drop (post-invisible test).

Prefix Redundancy



Definition: Index A contains (COL1, COL2) and Index B contains (COL1, COL2, COL3).

Remediation: Consolidate or rely on the larger index.

Uniqueness Redundancy



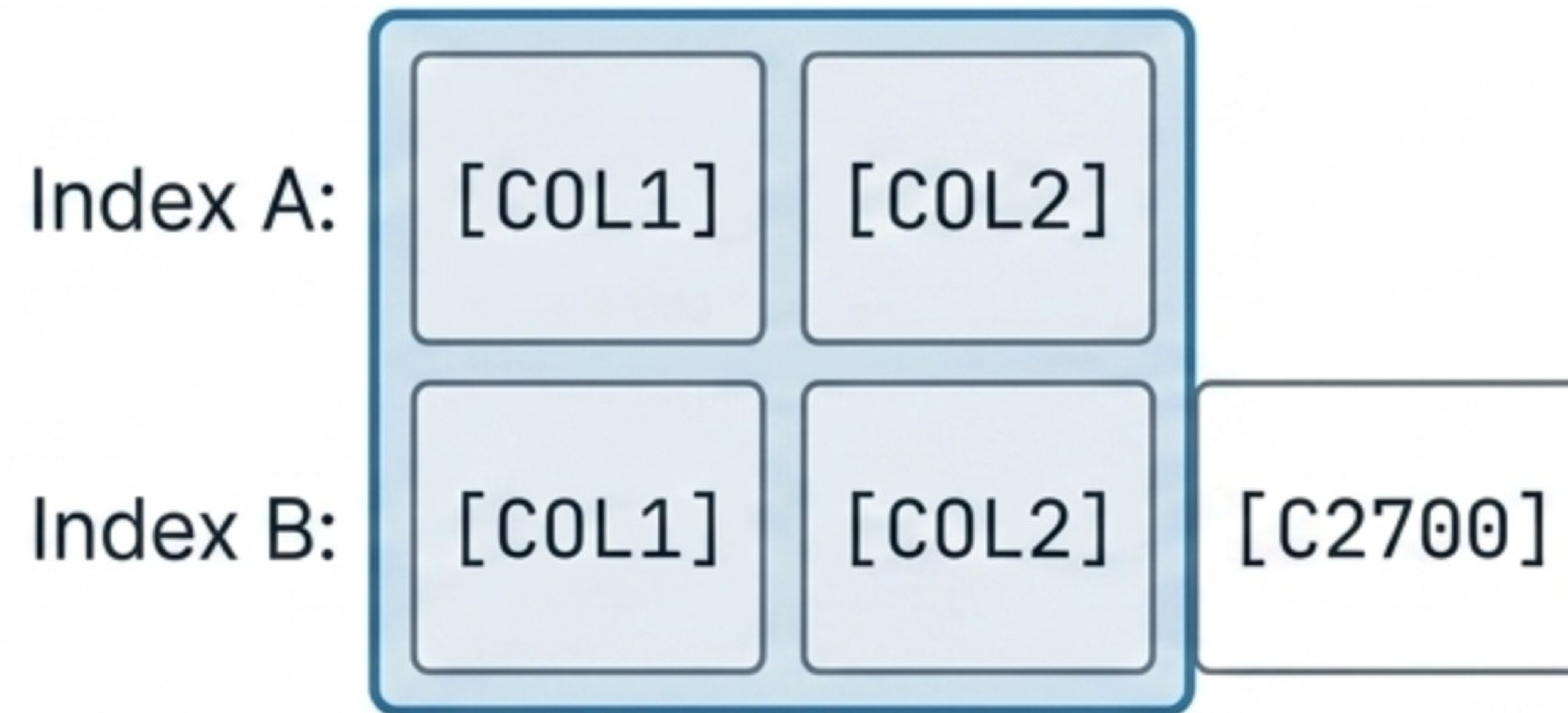
Definition: A Non-Unique index exists on the exact same column as a Unique constraint.

Remediation: Drop the manual index.

The Overlap Principle

Why Prefix Redundancy is deadly.

Index Subsumption Analysis

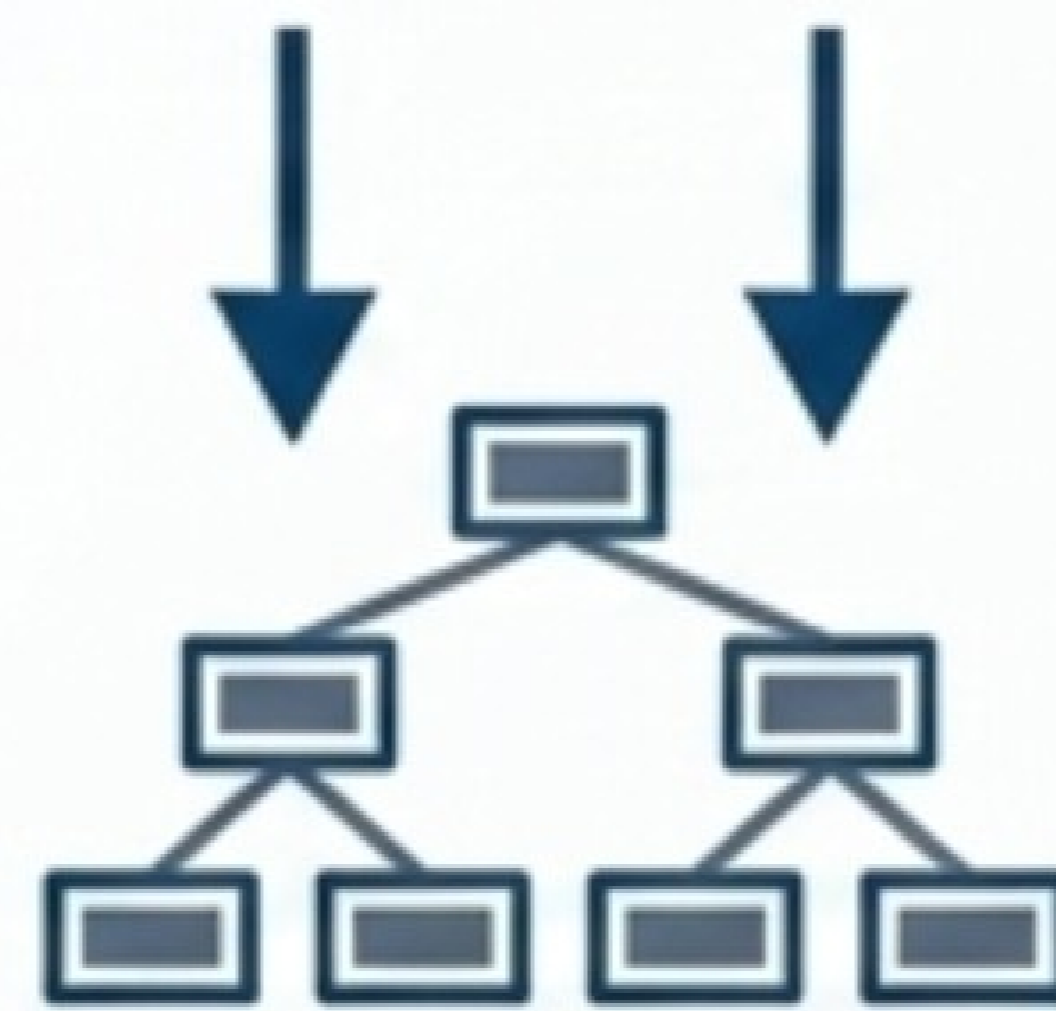


If Index B starts with everything in Index A, Index A is a ghost. It is entirely subsumed by the prefix of Index B. It cannot provide a unique access path, but it still eats space and DML cycles.

The Primary Key Pitfall



PRIMARY KEY
UNIQUE CONSTRAINT




~~CREATE INDEX idx_manual ON table_name(column);~~

Rule: Primary Keys and Unique Constraints automatically spawn their own B-Tree indexes in Oracle.
Actionable Advice: Never write a manual CREATE INDEX statement for columns that are already governed by a Primary Key or Unique Constraint. You are instantly creating Uniqueness Redundancy.

Diagnostic Telemetry: Scanning the Schema

Use the LISTAGG function on all_ind_columns to automatically group and flag identical column lists.

 Copy to Clipboard

```
SELECT table_name,  
       listagg(index_name, ',') WITHIN GROUP (ORDER BY index_name) as duplicate_indexes,  
       column_list  
FROM (  
  SELECT table_name, index_name,  
         listagg(column_name, ',') WITHIN GROUP (ORDER BY column_position) as column_list  
  FROM all_ind_columns WHERE table_owner = USER GROUP BY table_name, index_name  
)  
GROUP BY table_name, column_list HAVING COUNT(*) > 1;
```

Analyzing Telemetry Output

TABLE_NAME	DUPLICATE_INDEXES	COLUMN_LIST
CUSTOMERS	idx_cust_email	EMAIL_ADDRESS
SALES_DEMO	idx_cust_1, idx_cust_2, idx_cust_4	CUSTOMER_ID

The ``HAVING COUNT(*) > 1`` clause isolates exact overlaps. This output reveals three separate indexes maintaining the exact same ``CUSTOMER_ID`` path.

The Forensic Pruning Workflow

Step 1: Identify

Run the DBA_IND_COLUMNS overlap script to find redundant paths.

Step 2: Consolidate

Merge new column requirements into existing index prefixes rather than building net-new objects.

Step 3: Obscure

Deploy the INVISIBLE protocol to test the impact of removal safely.

Step 4: Prune

Drop the index only after a full business cycle validates stability.

The 'Invisible' Technique: The Ultimate Safety Net

ON

```
ALTER INDEX idx_name INVISIBLE;
```



Cost-Based Optimizer (Ignored)

The CBO acts as if the index has been dropped. It will immediately failover to alternative access paths.



DML Engine (Maintained)

The database continues to update the invisible index during INSERTS/UPDATES. If performance tanks, it can be toggled VISIBLE instantly without requiring a massive rebuild.

"If no one screams after a week, it's safe to drop."

Consolidation Strategy



The Anti-Pattern

```
Existing: Index_A (Region, Status)
New Requirement: Needs 'Date'
Action: CREATE INDEX Index_B on
(Region, Status, Date)
```

→ Creates prefix redundancy!



The Solution

```
Action: DROP INDEX Index_A;
CREATE INDEX Index_A_New on
(Region, Status, Date);
```

→ One index serves both query patterns.

System Directives & Best Practices



Always Scan First.

Never run `CREATE INDEX` without querying `DBA_IND_COLUMNS` to map existing architecture.



Trust Constraints.

Let Oracle automatically manage B-Trees for Primary Keys and Unique constraints. Do not duplicate them manually.



Obscure, Then Destroy.

Never drop an index blindly. Flag it `INVISIBLE`, monitor optimizer **response**, and drop only when proven safe.