From Minutes to Seconds: A Strategy to Fix Slow Queries

André Tipping

July 2025

Executive Summary: A Plan to Cut Pipeline Runtimes by 94% & Eliminate User Queuing

The Problem: The current warehouse configuration at this company is causing significant productivity loss. BI users are waiting over five minutes for dashboards to load, and critical data pipelines are delayed by inefficient processing.

Root Causes: These issues stem from a single source: a conflict between high-concurrency interactive BI workloads and memory-intensive automated data pipelines. Forcing them to compete creates bottlenecks for everyone.

The Solution: to isolate these workloads into dedicated, right-sized warehouses:

- **For BI Users:** Enable Multi-Cluster Warehousing to eliminate queuing and handle peak demand.
- **For Pipelines**: Create a new, dedicated TRANSFORM warehouse to provide the necessary memory and stop disk spillage.

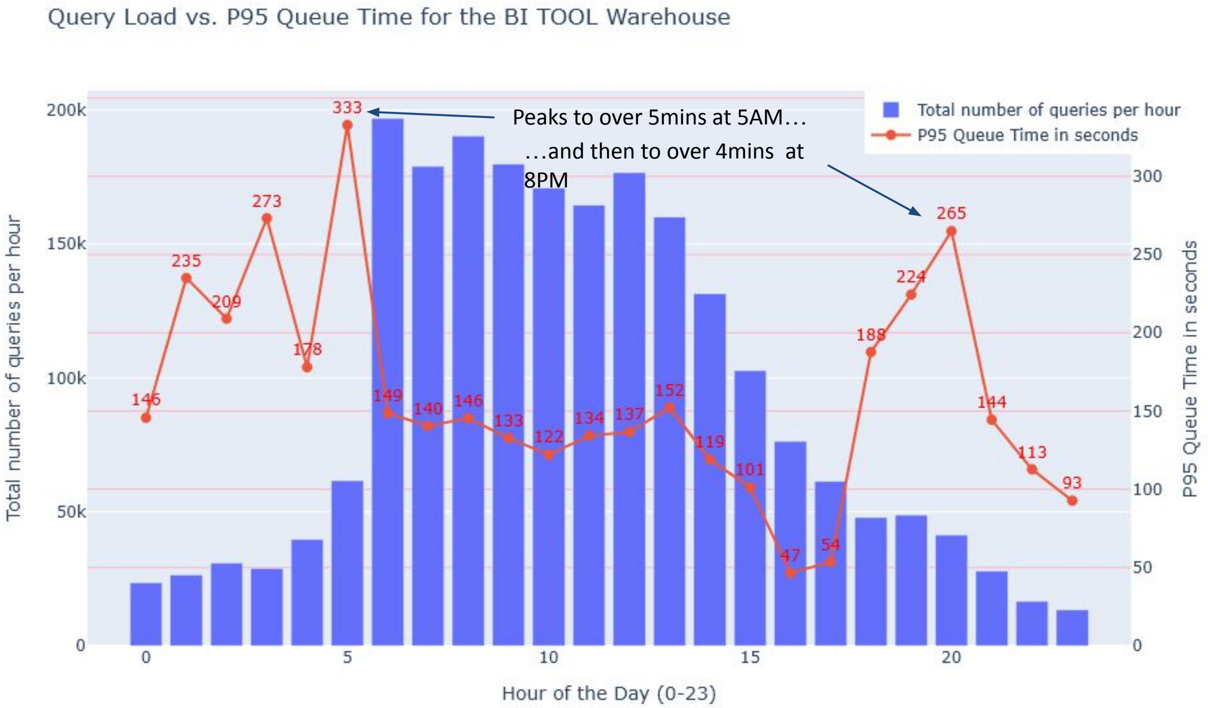
Outcome: For a modest net investment, a transformational improvement can be achieved. The result will be a fast, queue-free experience for BI users, reliable data pipelines that run up to 94% faster, and a more scalable and cost-effective data platform that can grow with the company.

The Root Cause: Conflicting Workloads

The data platform currently supports two primary workloads, and their performance needs are in direct conflict. An initial analysis confirmed that performance issues are concentrated in the interactive BI and automated ETL workloads, justifying the separate diagnosis that follows (see Appendix C).

Workload	Interactive (BI TOOL)	Automated Pipeline (SCHEDULER)
Needs	**High concurrency** to handle many simultaneous dashboard loads and user queries.	**High memory** for a few large, complex jobs.
Symptom of Failure	**Queuing**. Users are stuck in line. The BI TOOL WH has queue times that spike to over 5 minutes for just a few resource-intensive queries	**Spillage**. The warehouse runs out of memory. The SCHEDULER WH spills ~330TB of data to disk.
Impact	Lost user productivity and frustration.	Delayed data pipelines, risk of stale data, and high operational burden for engineers.

Diagnosis #1: BI Users Stuck in Queues



Query Load vs. P95 Queue Time for the BI TOOL Warehouse

Peaks to over 5mins at 5AM…

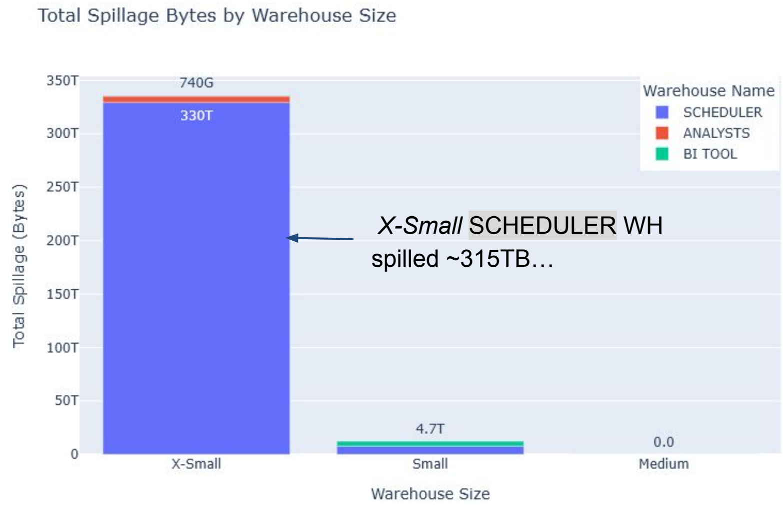
…and then to over 4mins at 8PM

Finding: The BI TOOL warehouse warehouse cannot handle the high volume of concurrent queries during peak hours, causing extreme queue times.

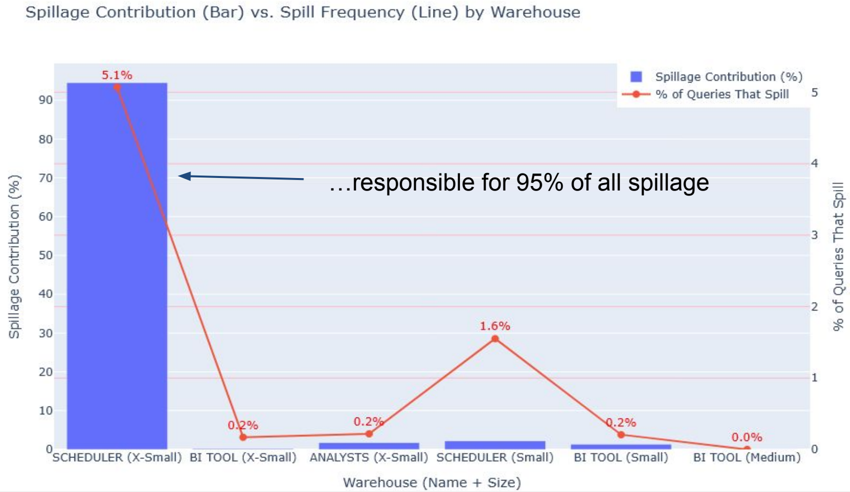
The Evidence: The 95th percentile queue time for the BI TOOL warehouse regularly exceeds 200 seconds.This happens even when the total number of queries isn't at its maximum, indicating a concurrency bottleneck where a few simultaneous, heavy queries block the queue for everyone else

The Impact: BI users are experiencing a slow, unresponsive platform, directly hurting productivity for over 103 peak hours in the last month (See Appx.G)

Diagnosis #2: Overwhelmed Undersized Warehouses


Total Spillage Bytes by Warehouse Size

X-Small SCHEDULER WH spilled ~315TB…


Spillage Contribution (Bar) vs. Spill Frequency (Line) by Warehouse

…responsible for 95% of all spillage

Finding: The *X-Small* SCHEDULER warehouse is fundamentally too small for its memory-intensive workload, causing massive disk spillage that slows pipelines to a crawl.

The Evidence:

- Over 5% of *X-Small* SCHEDULER's queries spill, and even with a *Small* config this value is at 1.6%.
- Automated CREATE_TABLE_AS_SELECT and INSERT queries are the primary culprits of this spillage (see Appendix D).
- Queries that spill to disk take 16 times longer to execute (Appx. F) , and these tend to be the most complex, automated jobs (Appx. E).
- This problem is getting worse, with spillage peaking at over 45TB in the final week of analysis (Appx. B).
- Heavy DELETE & MERGE jobs are causing locking contention, directly blocking BI user queries (Appx. I)

The Impact: This inefficiency means the data pipelines take 685 hours to complete when they could run in just 41. This creates a high risk of business decisions being made on stale data.

The Solution: A Dedicated Warehouse for Pipelines & Auto-Scaling for BI

The Strategy is to stop the conflict by creating separate, optimized environments for each workload.

1. **Tame Concurrency for BI:**
   a. **Action:** Enable Multi-Cluster Warehousing for the BI TOOL warehouse, scaling from 1 to 4 clusters. In addition it will be right-sized to a *Medium* size to prevent any spillage.
   b. **Benefit:** This will eliminate user queuing by automatically adding compute resources to handle high-concurrency loads, ensuring a fast, responsive BI experience.
2. **Solve Inefficiency for Pipelines:Action:**
   a. **Action**: Create a new, dedicated TRANSFORM warehouse (Size: *Medium*) and migrate all scheduled jobs to it.
   b. **Benefit:** This provides the necessary memory to eliminate disk spillage, making pipelines 94% faster and more reliable. It also prevents "noisy neighbor" problems by isolating these heavy jobs from our interactive users.

Business Case:
A 20% Cost Increase to Cut Pipeline Runtimes by 94%

Metric	**Current State (Inefficient)**	**Proposed State (Optimized)**
Monthly Cost	~1,239 Credits	~1,478 Credits
BI User Experience	Queues often exceed 5 minutes	Queuing eliminated, enabling faster decisions
Pipeline Performance	Jobs take ~7 minutes due to spillage	Jobs complete in ~26 seconds
Query Performance	Est. ~70% of queries < 30s	Est. ~90% of queries < 30s

A modest net increase of ~240 credits per month delivers a transformational improvement in platform performance. The investment gives over 100 peak hours back to BI users every month and makes critical data pipelines run 94% faster, slashing execution times from over 7 minutes to under 30 seconds.

Detailed cost projections (Appendix H) and performance estimates (Appendix J)

Next Steps: A Phased Approach to a Faster Platform

Phase 1: Immediate Infrastructure Fixes (Owner: Data Platform)

- Reconfigure the BI TOOL warehouse (Size: Medium, Multi-Clustering: On).
- Provision the new TRANSFORM warehouse (Size: Medium).

Phase 2: Workload Migration & Validation (Owner: Analytics Engineering)

- Systematically migrate all automated jobs to the TRANSFORM warehouse
- Prioritize & Optimize Locking Queries**:** As part of the migration, immediately investigate the long-running DELETE/MERGE queries on the REPORTS schema. Optimize these queries or reschedule them to off-peak hours to eliminate locking contention.
- Measure Success: Track P95 Queue Time (Target: <10s) and Total Spillage (Target: >95% reduction).

Phase 3: Building a Scalable Framework (Owner: AE, in partnership with key Data Consumers)

- Implement proactive monitoring and alerting for spillage and queuing.
- Implement Transaction Governance**:** Given the evidence of multi-day locks, develop a monitoring process to automatically detect and alert on long-running or idle transactions, preventing a single abandoned process from blocking major parts of the data platform in the future.
- Establish and document best practices for warehouse usage and query optimization.
- Map query costs to business teams to drive accountability and smarter data investments.

Thank you

Questions?

Appendix A: Methodology & Approach

Data Sources & Preparation

- Ingested query performance logs (log_perf) and object access logs (log_objs) of over 10M rows each.
- Joined the two datasets on QUERY_ID to link performance metrics with query context, covering the period from March 28 to June 28 2021.
- Engineered key features for analysis:
    - schema_count: To serve as a proxy for query complexity
    - Time-based features (hour, day_of_week) to analyze usage patterns.
    - Blocked Time: Total Time - (Execution + Queue Time)

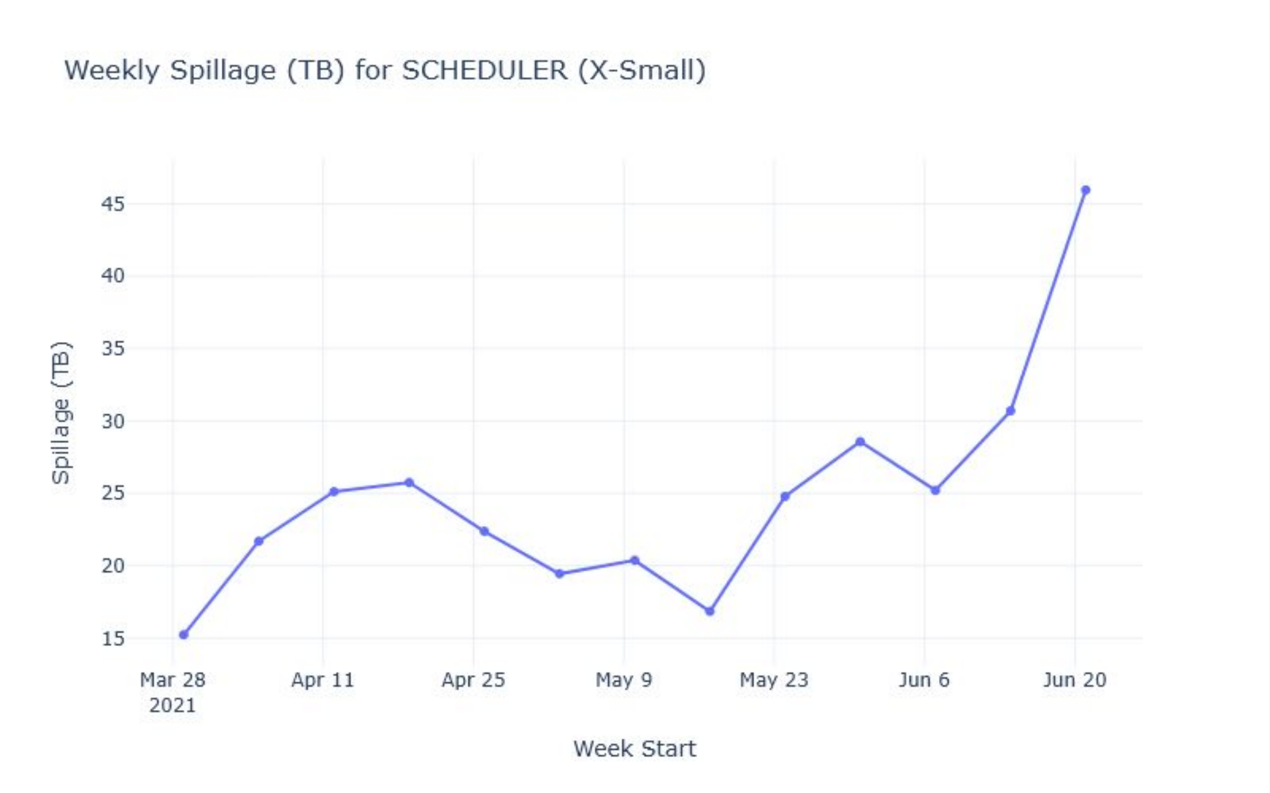
Analytical Framework

- The analysis was structured to understand the system's behaviour & then pinpoint the root cause of the slowdown.
    - **Workload Characterization:** First, I profiled the platform's distinct workloads (e.g., interactive BI, scheduled ETL) by analyzing their unique query signatures and resource needs.
    - **Diagnose Bottlenecks:** I diagnosed the specific failure point for each workload, focusing on concurrency contention (queuing) for interactive users and resource exhaustion (disk spillage) for batch jobs.
    - **Synthesize the Root Cause:** I synthesized these findings to identify the core problem: a platform-wide slowdown caused by the conflict between these improperly isolated workloads.

Tools Used

- Data processing performed using Python with Pandas/Polars.
- Visualizations created with Plotly.
- Analysis conducted in a Jupyter notebook.

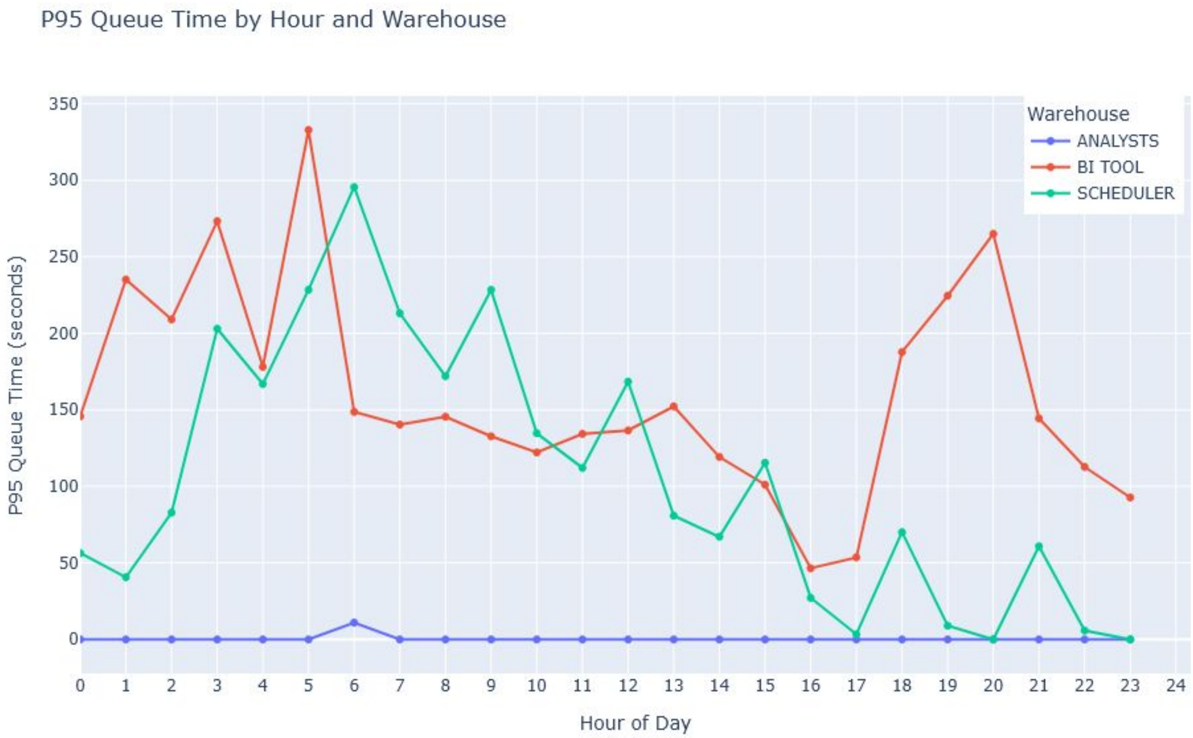
Appendix B: Weekly Spillage for X-Small SCHEDULER



Key Insight: The SCHEDULER (X-Small) WH is consistently spilling over 15TB; and has been growing in the last three weeks to peak at > 45TB.

Implication: The performance of the data pipelines is actively deteriorating, increasing the risk of stale data each week. This creates urgency to move these jobs to a right-sized warehouse before they fail completely.

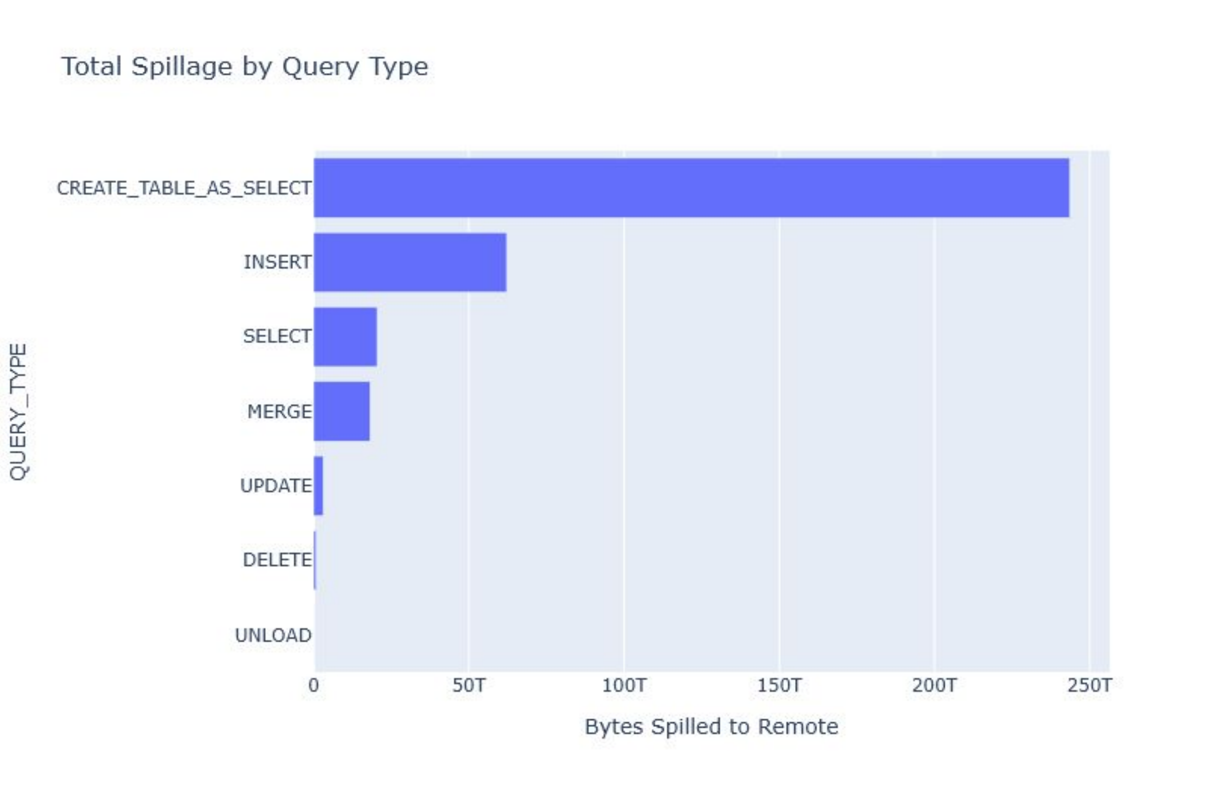
Appendix C: Queuing Analysis Across All Warehouses



Insight: An overview of P95 queue times shows that both the BI TOOL and SCHEDULER warehouses experience significant queuing, while the ANALYSTS warehouse has no contention issues.

Implication: This confirms that the performance problems are concentrated in interactive BI and automated ETL workloads. This justifies the deeper, separate diagnosis for each of these two areas to understand their distinct root causes.

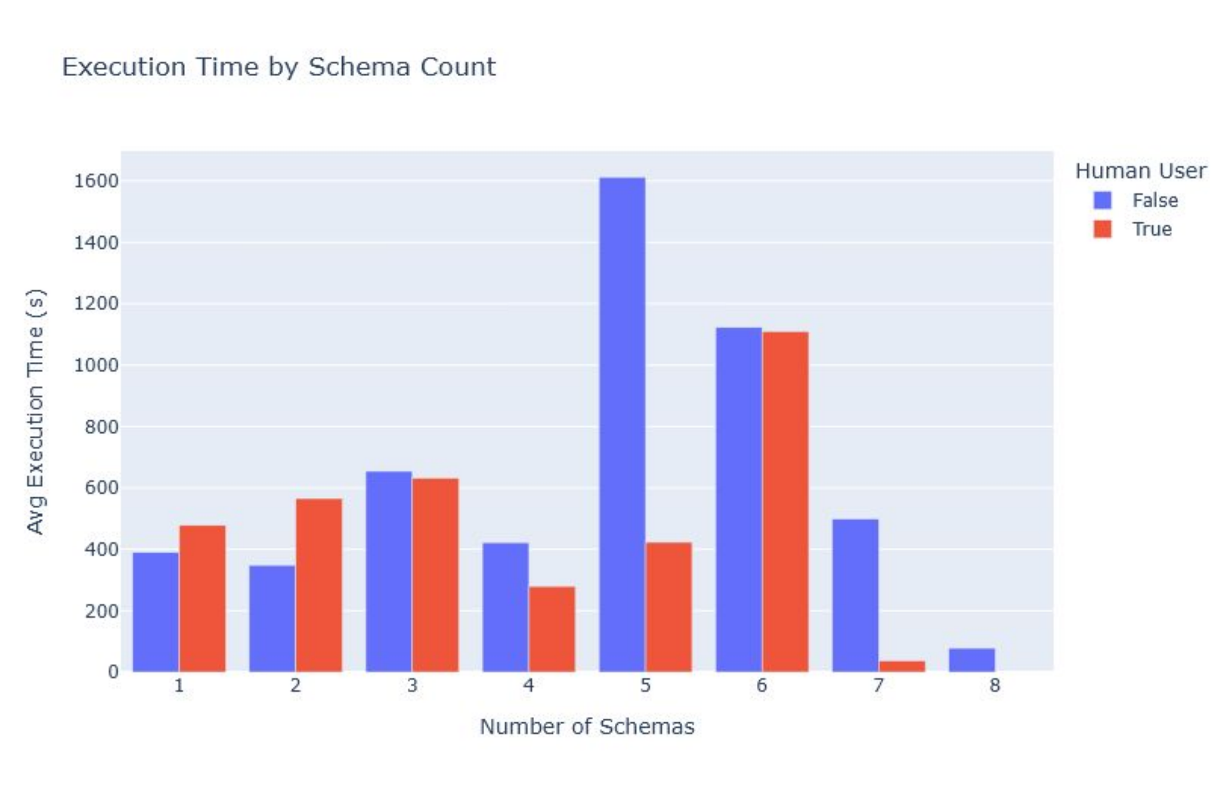
Appendix D: Total Spillage by Query Type



Key Insight: CTAS queries dwarf all other types accounting for nearly 250 TB

Implication: The problem is highly concentrated in core ETL processes. This allows the solution to be very targeted, by isolating just these heavy write operations, the vast majority of the inefficiency on the platform can be solved for.

Appendix E: Execution Time by Schema Count



Key Insight: Insight: Average query execution time for automated jobs rises dramatically as the number of schemas joined increases.

Implication: As business questions and data models become more complex, the current infrastructure will increasingly fail. The proposed changes are a necessary investment to support future analytical needs.

Appendix F: Impact of Spillage on Execution Time

Query Performance on the SCHEDULER (X-Small) WH

Has Queue	Has Spillage	Avg Execution Time (s)	Number of Queries
True	True	530	26467
False	True	432	12896
True	False	91	160834
False	False	26	576246

Key Insight: The SCHEDULER (X-Small) WH takes 16x longer to execute queries when spilling to disk than those that do not.

Implication: This quantifies the immense cost of using an undersized warehouse. Spillage is not a minor issue; it's a catastrophic performance failure that proves investing in a larger warehouse will have an immediate and dramatic ROI on pipeline speed.

Appendix G: 30-Day Peak Hour Analysis

WH Name	WH Size	peak_status	total_hours
BI TOOL	X-Small	peak	421
BI TOOL	Small	peak	103
BI TOOL	X-Small	off-peak	68
BI TOOL	Small	off-peak	174
SCHEDULER	X-Small	off-peak	368
SCHEDULER	Small	off-peak	63
SCHEDULER	Small	peak	1
SCHEDULER	X-Small	peak	317

Peak Hour is defined as hours when 95th percentile queue time exceeded 60 seconds

Key Insight: The BI TOOL warehouse is the primary source of user-facing queuing, experiencing over 520 peak hours in the last 30 days.

Implication: The maximum positive impact to the largest number of users can be delivered by focusing on fixing the concurrency problem for this single warehouse.

Appendix H: Business Case Calculations

Current State: Inefficient & Inexpensive (Last 30 Days)

BI TOOL (Small Warehouse):

- Active for 277 hrs (103 peak, 174 off-peak Appx.G)
- 30D Cost: 277 * 2 credits/hr = 554 Credits

SCHEDULER (X-Small Warehouse):

- Active for 685 hrs (317 peak, 368 off-peak).
- 30-Day Cost: 685 hrs * 1 credit/hr = 685 Credits

Total 30-Day Cost for Problem Workloads: **1,239 Credits**

Proposed State: A Strategic Investment in Performance

BI TOOL (Medium Warehouse w/ Multi-clustering):

- Base Cost: 277 hrs * 4 credits/hr = 1,108 credits.
- Multi-cluster Premium: (Est. 50% for 103 peak hrs): 103 hrs * (4 credits/hr * 50%) = 206 credits..
- **Projected** 30-Day Cost: 1,314 Credits

TRANSFORM (New Medium Warehouse for Scheduled Jobs):

- Estimated 94% execution time reduction by eliminating spillage (Appx. F).
- Projected Hours: 685 hr * (1 - 0.94) = ~41 hrs.
- **Projected** 30-Day Cost: 41 hrs * 4 credits/hour = 164 Credits

Total Projected 30-Day Cost: **1,478 Credits**

Appendix I: Locking Chain Analysis

Victim Warehouse	Victim Blocked Time (sec)	Culprit Query Type	Shared Schema
BI TOOL	1,280,489	DELETE	REPORTS
BI TOOL	1,177,798	MERGE	REPORTS
ANALYSTS	379,362	DELETE	REPORTS

Key Insight: The analysis identified multiple queries that were blocked for several days due to locks held by DELETE and MERGE statements.

Implication: This transcends a simple performance issue and points to a critical operational risk of unmanaged or abandoned transactions. The solution must therefore include not just infrastructure changes but also improved governance and monitoring to prevent platform-wide stalls.

Appendix J: Performance Projection Methodology

To estimate the impact of the proposed solutions, the change in the percentage of queries completing in under 30 seconds was projected. This was based on the specific performance gains identified.

1. Simulation Logic

- **For BI TOOL Queries:** The PROJECTED_TOTAL_TIME was calculated by removing the QUEUE_TIME, simulating the effect of enabling multi-clustering.
- **For Spilling SCHEDULER Queries:** The EXECUTION_TIME was replaced with the median execution time of non-spilling SCHEDULER queries. The QUEUE_TIME was also removed to simulate the effect of moving to an isolated warehouse.
- **For All Other Queries:** The TOTAL_TIME remained unchanged.

2. Key Calculation Inputs & Results

- Current % of Queries Completing < 30s: 73.22%
- Projected % of Queries Completing < 30s: 89.70%