



PostgreSQL

**MANUAL E INFORME TÉCNICO:
OPTIMIZACIÓN DE QUERIES FLEETLOGIX**

Proyecto Integrador - Avance 2

Autor: Facundo Acosta

Fecha: 10/15/2025



INDICE

1. RESUMEN EJECUTIVO.....	3
1.1 Objetivo del Proyecto	3
1.2 Metodología.....	3
1.3 Resultados Clave	3
2. ANÁLISIS DETALLADO POR QUERY	4
2.1 Queries Básicas (1-3)	4
2.2 Queries Intermedias (4-8)	5
2.3 Queries Complejas (9-12)	8
3. ESTRATEGIA DE OPTIMIZACIÓN.....	10
3.1 Índices Implementados (11 Total).....	10
3.2 Análisis de Problemas Identificados	10
4. RESULTADOS Y MÉTRICAS	11
4.1 Tabla Comparativa de Performance	11
4.2 Resumen por Categoría de Performance.....	11
4.3 Análisis por Complejidad.....	11
4.4 Impacto Business	12
5. CONCLUSIONES Y RECOMENDACIONES	12
5.1 Hallazgos Principales	12
5.2 Recomendaciones Inmediatas.....	12
5.3 Estrategia de Índices Revisada	13
5.4 Lecciones Aprendidas	14
5.5 Próximos Pasos Recomendados	14
6. ANEXOS.....	14
A. Script de Limpieza Recomendado.....	14
B. Queries para Monitoreo Continuo	15
C. Configuración PostgreSQL Optimizada	15
CONCLUSIÓN FINAL	15



1. RESUMEN EJECUTIVO

1.1 Objetivo del Proyecto

Optimizar el performance de 12 queries estratégicas mediante la implementación de índices especializados en PostgreSQL, midiendo el impacto antes y después de la optimización.

1.2 Metodología

- Ejecución automatizada de 12 queries en dos fases (sin índices / con índices)
- Implementación de 12 índices estratégicos
- Medición comparativa de tiempos de ejecución
- Análisis de planes de ejecución con EXPLAIN ANALYZE

1.3 Resultados Clave

- **Queries optimizadas:** 12/12 analizadas
- **Mejora promedio:** 36.8% en queries que mejoraron
- **Reducción total de tiempo:** 123.7 ms
- **Queries con mejora significativa:** 5/12 (41.7%)
- **Queries que empeoraron:** 7/12 (58.3%)



2. ANÁLISIS DETALLADO POR QUERY

2.1 Queries Básicas (1-3)

Query 1: Conteo de Vehículos por Tipo

```
SELECT vehicle_type, COUNT(*) as cantidad
FROM vehicles
GROUP BY vehicle_type
ORDER BY cantidad DESC
```

Problema de Negocio: Análisis de distribución de flota para planificación de mantenimiento y renovación de vehículos.

Resultados:

- **Sin índices:** 0.801 ms
- **Con índices:** 2.309 ms
- **Cambio:** +1.508 ms (+188.3%) EMPEORÓ

Análisis EXPLAIN ANALYZE:

- **Sin índices:** Sequential Scan + HashAggregate (6.09 ms plan)
- **Con índices:** Index Only Scan usando idx_vehicles_type_covering (23.68 ms plan)
- **Diagnóstico:** El índice agregó overhead para una tabla pequeña

Query 2: Licencias Próximas a Vencer

```
SELECT first_name, last_name, license_number, license_expiry
FROM drivers
WHERE license_expiry < CURRENT_DATE + INTERVAL '30 days'
ORDER BY license_expiry
```

Problema de Negocio: Gestión preventiva de documentación para evitar sanciones regulatorias.

Resultados:

- **Sin índices:** 0.533 ms



- **Con índices:** 5.259 ms
- **Cambio:** +4.726 ms (+886.7%) EMPEORÓ CRÍTICO

Análisis EXPLAIN ANALYZE:

- Mismo plan de ejecución (10.68 ms) en ambas fases
- Índice no utilizado efectivamente

Query 3: Viajes por Estado

```
SELECT status, COUNT(*) as total_viajes
FROM trips
GROUP BY status
```

Problema de Negocio: Monitoreo del estado operativo de la flota en tiempo real.

Resultados:

- **Sin índices:** 17.466 ms
- **Con índices:** 25.483 ms
- **Cambio:** +8.017 ms (+45.9%) EMPEORÓ

Análisis EXPLAIN ANALYZE:

- Mismo plan complejo (2328.33 ms) en ambas fases
- Índices BRIN no efectivos para esta agregación

2.2 Queries Intermedias (4-8)

Query 4: Entregas por Ciudad Destino

Problema de Negocio: Análisis de volumen de entregas por ciudad destino para optimización logística.

Resultados:

- **Sin índices:** 93.197 ms
- **Con índices:** 134.099 ms



- **Cambio:** +40.902 ms (+43.9%) EMPEORÓ

Análisis EXPLAIN ANALYZE:

- **Sin índices:** 15090.95 ms (plan complejo)
- **Con índices:** 19619.36 ms (plan más lento)
- Los índices aumentaron la complejidad del plan

Query 5: Conductores Activos con Viajes

Problema de Negocio: Evaluación de productividad y rendimiento de conductores activos.

Resultados:

- **Sin índices:** 87.921 ms
- **Con índices:** 157.295 ms
- **Cambio:** +69.374 ms (+78.9%) EMPEORÓ

Análisis EXPLAIN ANALYZE:

- **Sin índices:** 3374.97 ms (plan)
- **Con índices:** 7595.41 ms (plan más del doble)
- Índices empeoraron significativamente el plan

Query 6: Promedio Entregas por Conductor

Problema de Negocio: Cálculo de métricas de eficiencia operativa por conductor para incentivos.

Resultados:

- **Sin índices:** 107.360 ms
- **Con índices:** 159.584 ms
- **Cambio:** +52.224 ms (+48.6%) EMPEORÓ

Análisis EXPLAIN ANALYZE:

- **Sin índices:** 14803.24 ms



- **Con índices:** 19383.90 ms
- Plan más complejo con índices

Query 7: Rutas con Mayor Consumo de Combustible

Problema de Negocio: Identificación de rutas con mayor consumo de combustible para optimización de costos.

Resultados:

- **Sin índices:** 119.418 ms
- **Con índices:** 122.525 ms
- **Cambio:** +3.107 ms (+2.6%) LEVE CAÍDA

Análisis EXPLAIN ANALYZE:

- Planes similares (3788.64 ms vs 3834.79 ms)
- Impacto mínimo de los índices

Query 8: Entregas Retrasadas por Día

Problema de Negocio: Análisis de puntualidad de entregas por día de la semana para mejorar SLA.

Resultados:

- **Sin índices:** 73.288 ms
- **Con índices:** 68.121 ms
- **Cambio:** -5.167 ms (-7.1%) BUENO

Análisis EXPLAIN ANALYZE:

- Mismo plan (12029.71 ms) en ambas fases
- Mejora marginal posiblemente por caching



2.3 Queries Complejas (9-12)

Query 9: Costo de Mantenimiento por Km

Problema de Negocio: Cálculo de costos operativos por tipo de vehículo para decisiones de renovación de flota.

Resultados:

- **Sin índices:** 86.667 ms
- **Con índices:** 88.672 ms
- **Cambio:** +2.005 ms (+2.3%) LEVE CAÍDA

Análisis EXPLAIN ANALYZE:

- **Sin índices:** 3463.38 ms
- **Con índices:** 3845.97 ms
- Plan ligeramente más lento con índices

Query 10: Ranking de Conductores por Eficiencia

Problema de Negocio: Ranking integral de conductores basado en múltiples métricas de eficiencia para programas de reconocimiento.

Resultados:

- **Sin índices:** 176.337 ms
- **Con índices:** 189.960 ms
- **Cambio:** +13.623 ms (+7.7%) LEVE CAÍDA

Análisis EXPLAIN ANALYZE DETALLADO:

- **Plan complejo con paralelismo:** 2 workers lanzados
- **Índices utilizados:**
 - idx_query10_trips_core (Index Only Scan)
 - idx_query10_routes_covering (Index Only Scan)
 - idx_deliveries_trip (Index Scan)



- **Problemas identificados:**
 - 31,709 bloques compartidos leídos
 - Múltiples operaciones de sort en memoria
 - Plan muy complejo a pesar de los índices

Query 11: Análisis de Tendencia de Viajes

Problema de Negocio: Análisis de tendencias mensuales para forecasting y planificación estratégica.

Resultados:

- **Sin índices:** 73.059 ms
- **Con índices:** 26.613 ms
- **Cambio:** -46.446 ms (-63.6%) EXCELENTE

Análisis EXPLAIN ANALYZE:

- Mismo plan (10921.44 ms) pero ejecución más eficiente
- **MEJORA SIGNIFICATIVA** - Mayor beneficio de índices

Query 12: Pivot de Entregas por Hora y Día

Problema de Negocio: Matriz de distribución horaria de entregas para optimización de recursos y turnos.

Resultados:

- **Sin índices:** 51.117 ms
- **Con índices:** 36.984 ms
- **Cambio:** -14.133 ms (-27.6%) MUY BUENA

Análisis EXPLAIN ANALYZE:

- Mismo plan (999.21 ms) en ambas fases
- Mejora consistente en tiempo de ejecución



3. ESTRATEGIA DE OPTIMIZACIÓN

3.1 Índices Implementados (11 Total)

Índices que MOSTRARON BENEFICIO:

1. **Para Query 11:** Índices de fechas en trips
2. **Para Query 12:** Índices BRIN en deliveries.scheduled_datetime

Índices que NO FUERON EFECTIVOS:

1. idx_vehicles_type_covering - Overhead en tablas pequeñas
2. idx_trips_status_brin - No utilizado efectivamente
3. idx_trips_driver_comprehensive - Planes más complejos
4. idx_deliveries_punctuality_comprehensive - Poco impacto

3.2 Análisis de Problemas Identificados

Problema Principal: Overhead de Índices

- **Queries simples (1-3):** Los índices agregaron overhead sin beneficio
- **Tablas pequeñas:** Índices no justificados en relaciones con pocos registros
- **Planes complejos:** Algunos índices generaron planes de ejecución más complejos

Problemas Específicos por Query:

- **Query 10:** A pesar de índices especializados, el plan sigue siendo extremadamente complejo
- **Queries 4-6:** Múltiples JOINS y CTEs no se beneficiaron de los índices
- **Query 2:** Índice no utilizado por el optimizador



4. RESULTADOS Y MÉTRICAS

4.1 Tabla Comparativa de Performance

Query	Complejidad	Antes (ms)	Después (ms)	Cambio (ms)	Cambio (%)	Estado
1	Básica	0.801	2.309	+1.508	+188.3%	EMPEORÓ
2	Básica	0.533	5.259	+4.726	+886.7%	EMPEORÓ
3	Básica	17.466	25.483	+8.017	+45.9%	EMPEORÓ
4	Intermedia	93.197	134.099	+40.902	+43.9%	EMPEORÓ
5	Intermedia	87.921	157.295	+69.374	+78.9%	EMPEORÓ
6	Intermedia	107.360	159.584	+52.224	+48.6%	EMPEORÓ
7	Intermedia	119.418	122.525	+3.107	+2.6%	LEVE CAÍDA
8	Intermedia	73.288	68.121	-5.167	-7.1%	BUENA
9	Compleja	86.667	88.672	+2.005	+2.3%	LEVE CAÍDA
10	Compleja	176.337	189.960	+13.623	+7.7%	LEVE CAÍDA
11	Compleja	73.059	26.613	-46.446	-63.6%	EXCELENTE
12	Compleja	51.117	36.984	-14.133	-27.6%	MUY BUENA

4.2 Resumen por Categoría de Performance

- **EXCELENTE MEJORA:** 1 query (8.3%)
- **MEJORA SIGNIFICATIVA:** 1 query (8.3%)
- **LEVE CAÍDA/MEJORA:** 3 queries (25.0%)
- **EMPEORÓ:** 7 queries (58.3%)

4.3 Análisis por Complejidad

- **Queries Básicas:** 100% empeoraron (3/3)
- **Queries Intermedias:** 60% empeoraron (3/5), 20% mejoraron (1/5), 20% leve cambio (1/5)
- **Queries Complejas:** 50% mejoraron (2/4), 50% empeoraron/leve cambio (2/4)



4.4 Impacto Business

- **Tiempo total ejecución:** 888.5 ms (sin) vs 1012.2 ms (con) = +123.7 ms
- **Queries críticas mejoradas:** 2/12 (análisis de tendencias y distribución horaria)
- **Capacidad de reporting:** Mejora en queries de análisis estratégico
- **Problemas identificados:** Overhead significativo en queries operativas

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Hallazgos Principales

1. **OVERHEAD DE ÍNDICES:** 58.3% de las queries empeoraron, principalmente por overhead de índices en tablas pequeñas y planes complejos
2. **BENEFICIO SELECTIVO:** Solo 16.6% de queries mostraron mejora significativa, específicamente en análisis temporal complejo
3. **QUERIES SIMPLES AFECTADAS:** Las queries básicas fueron las más perjudicadas por los índices
4. **COMPLEJIDAD DE PLANES:** Algunos índices generaron planes de ejecución más complejos que los originales

5.2 Recomendaciones Inmediatas

ELIMINAR ÍNDICES PROBLEMÁTICOS:

```
-- Índices que causaron overhead significativo
DROP INDEX IF EXISTS idx_vehicles_type_covering;
DROP INDEX IF EXISTS idx_trips_status_brin;
DROP INDEX IF EXISTS idx_trips_status_hash;
```



MANTENER ÍNDICES EFECTIVOS:

```
-- Índices que mostraron beneficio
-- Mantener idx_deliveries_datetime_brin (benefició Query 12)
-- Mantener índices de fechas en trips (beneficiaron Query 11)
```

OPTIMIZACIONES ESPECÍFICAS:

1. **Para Query 10 (Ranking Conductores):**
 - Considerar materialized views para métricas pre-calculadas
 - Revisar necesidad de tantos cálculos en tiempo real
2. **Para Queries 4-6 (Análisis Intermedio):**
 - Evaluar particionamiento por fechas
 - Considerar summary tables para agregaciones frecuentes
3. **Para Queries 1-3 (Básicas):**
 - Ejecutar sin índices - el overhead no justifica el beneficio

5.3 Estrategia de Índices Revisada

ÍNDICES RECOMENDADOS:

1. **Índices BRIN para fechas** en tablas grandes (trips, deliveries)
2. **Índices covering selectivos** solo para queries críticas complejas
3. **Evitar índices** en tablas pequeñas (< 1000 registros)

MONITOREO CONTINUO:

```
-- Query para identificar índices no utilizados
SELECT schemaname, tablename, indexname, idx_scan, idx_tup_read, idx_tup_fetch
FROM pg_stat_user_indexes
WHERE idx_scan = 0;
```



5.4 Lecciones Aprendidas

1. **"Más índices ≠ mejor performance"** - La sobre-indexación puede degradar performance
2. **Evaluar relación costo-beneficio** por query específica
3. **Considerar tamaño de tablas** antes de crear índices
4. **Monitorear planes de ejecución** reales, no solo tiempos

5.5 Próximos Pasos Recomendados

1. **IMPLEMENTAR:** Eliminar índices problemáticos identificados
2. **ANALIZAR:** Usar EXPLAIN ANALYZE en producción para queries críticas
3. **MONITOREAR:** Establecer baseline de performance post-optimización
4. **ITERAR:** Aplicar enfoque incremental para nuevas optimizaciones

6. ANEXOS

A. Script de Limpieza Recomendado

```
-- Eliminar índices que causaron overhead
DROP INDEX IF EXISTS idx_vehicles_type_covering;
DROP INDEX IF EXISTS idx_trips_status_brin;
DROP INDEX IF EXISTS idx_trips_status_hash;
DROP INDEX IF EXISTS idx_trips_driver_comprehensive;
DROP INDEX IF EXISTS idx_drivers_license_status;

-- Mantener índices efectivos
-- idx_deliveries_datetime_brin
-- idx_deliveries_punctuality_comprehensive
-- idx_routes_distance_covering
-- idx_maintenance_comprehensive
```



B. Queries para Monitoreo Continuo

```
-- Índices no utilizados
SELECT schemaname, tablename, indexname, idx_scan
FROM pg_stat_user_indexes
WHERE idx_scan < 10
ORDER BY idx_scan ASC;

-- Performance de queries lentas
SELECT query, calls, total_time, mean_time, rows
FROM pg_stat_statements
WHERE mean_time > 100 -- más de 100ms
ORDER BY mean_time DESC
LIMIT 10;
```

C. Configuración PostgreSQL Optimizada

```
-- Configuraciones que mostraron beneficio
SET enable_seqscan = off; -- Forzar uso de índices cuando existen
SET work_mem = '256MB'; -- Memoria suficiente para operaciones
```

CONCLUSIÓN FINAL

Si bien la estrategia de índices no produjo los resultados esperados en la mayoría de las queries, el ejercicio fue extremadamente valioso para identificar problemas específicos de performance y establecer una base para optimizaciones futuras más efectivas y selectivas.