Starburst

From Kafka to insights: high-throughput ingestion at Starburst

100GB/s and Beyond

Lakshmikant (Pachu) Shrinivas

Feb 18, 2026

Pachu

Staff Software Engineer

Data ingestion is just a fact of life

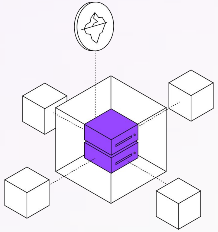
Ingestion tailored for Iceberg data lakes

Starburst

Starburst is a data platform built on a data lakehouse architecture with Trino + Apache Iceberg and designed to accelerate analytic workflows from development to deployment.

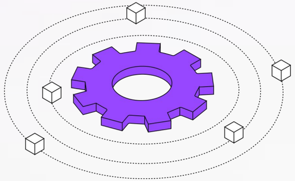
For an end-to-end data warehouse experience, five core capabilities must exist

Starburst Galaxy Platform



Data Ingestion

How data lands in the lake

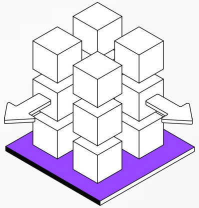


Query Engine

How data is processed for data pipelines and analyzed for consumption

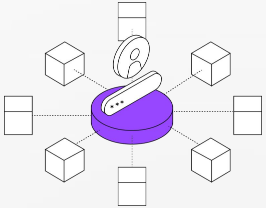
Data Management

How tables are cleaned and optimized to maintain performance



Automatic Capacity Management

How compute scales up and down to match demand



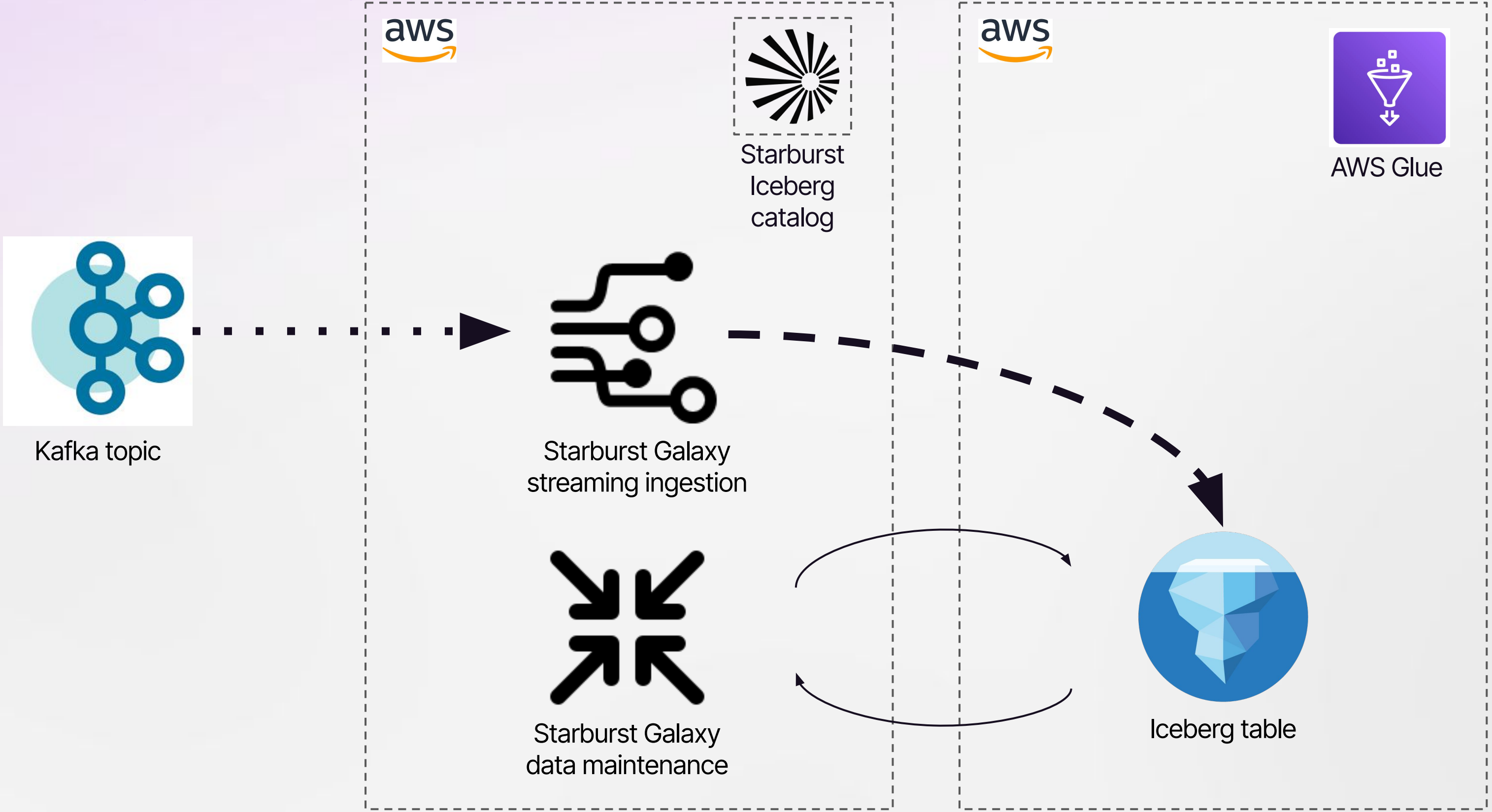
Data Governance

How data is accessed and secured once in the lake

Starburst

Simplified Kafka-to-Iceberg ingestion

Starburst ingestion: designed for Iceberg


aws
Starburst Iceberg catalog
aws
AWS Glue
Kafka topic
Starburst Galaxy streaming ingestion
Starburst Galaxy data maintenance
Iceberg table

Starburst

Unlocking Iceberg streaming ingestion

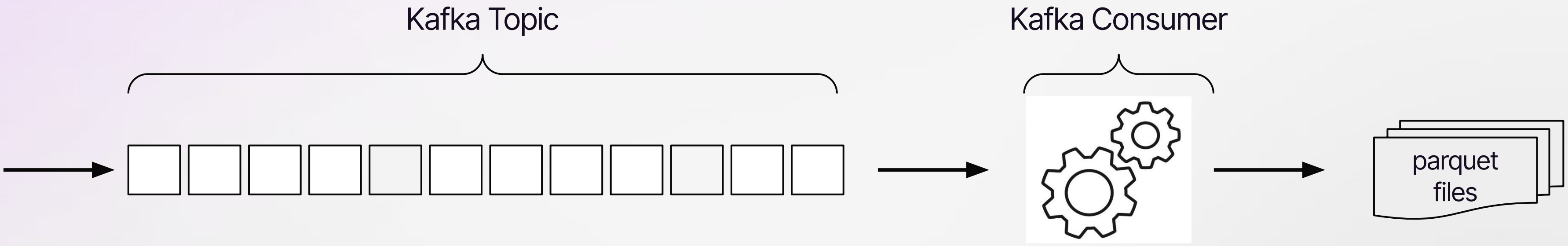
Building for Scale

Planning for Imperfection

Unified Data Platform

Starburst

Kafka scaling 101

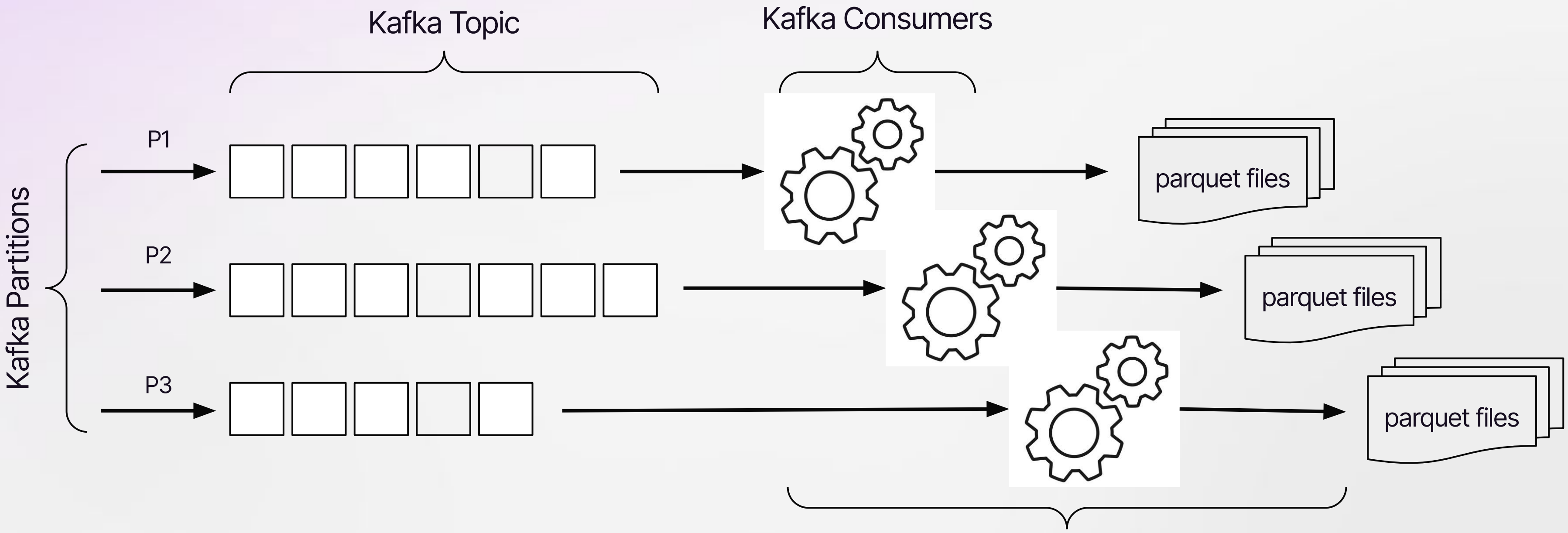


Performance measured in data throughput

e.g. 1 MBps

Starburst

Kafka partitioning increases throughput

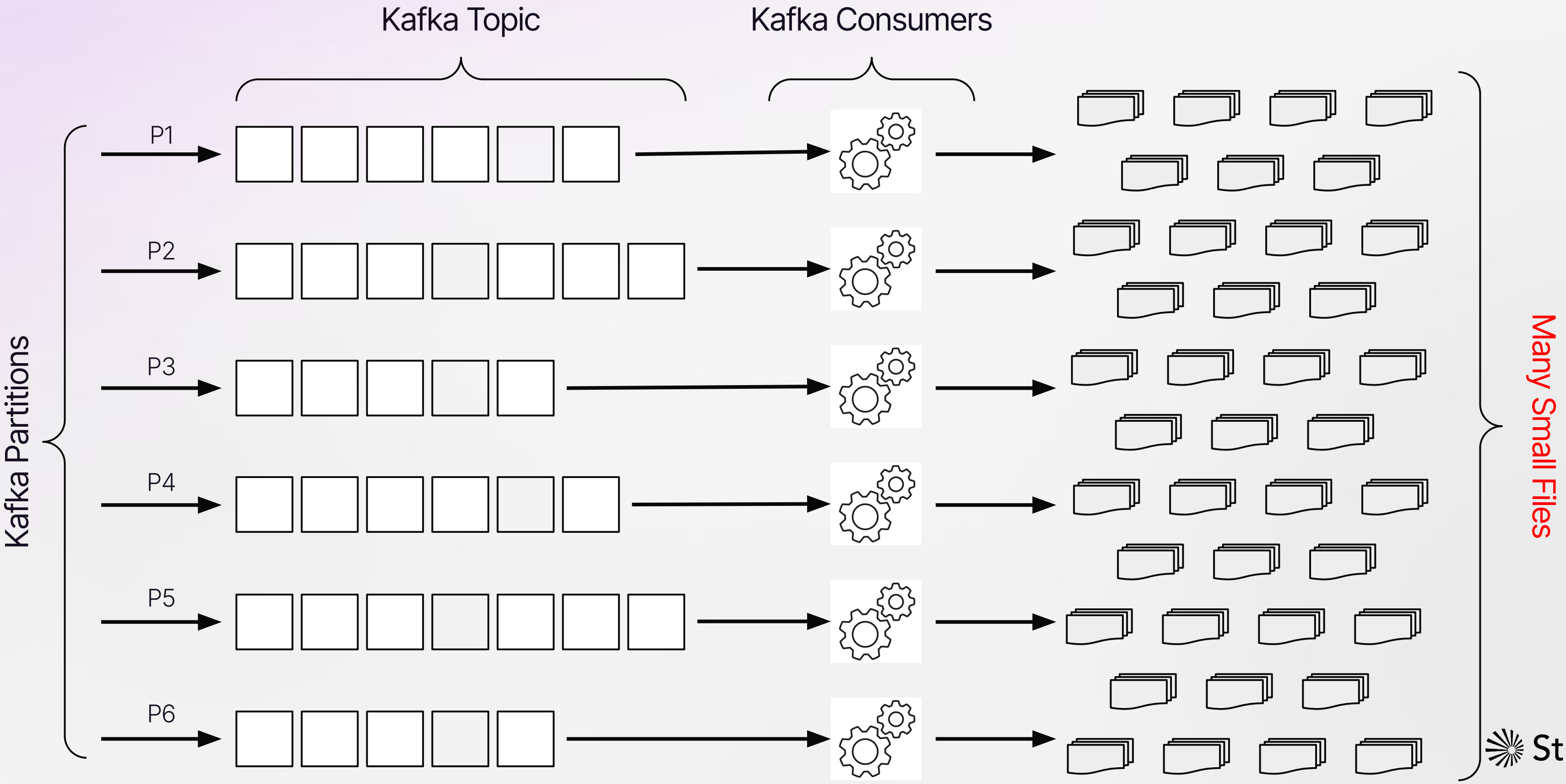


Performance measured in total data throughput
e.g. 1 MBps / partition * 3 partitions = 3 MBps

Starburst

Are we done with data scaling concerns?

Challenge - parallelism vs compaction

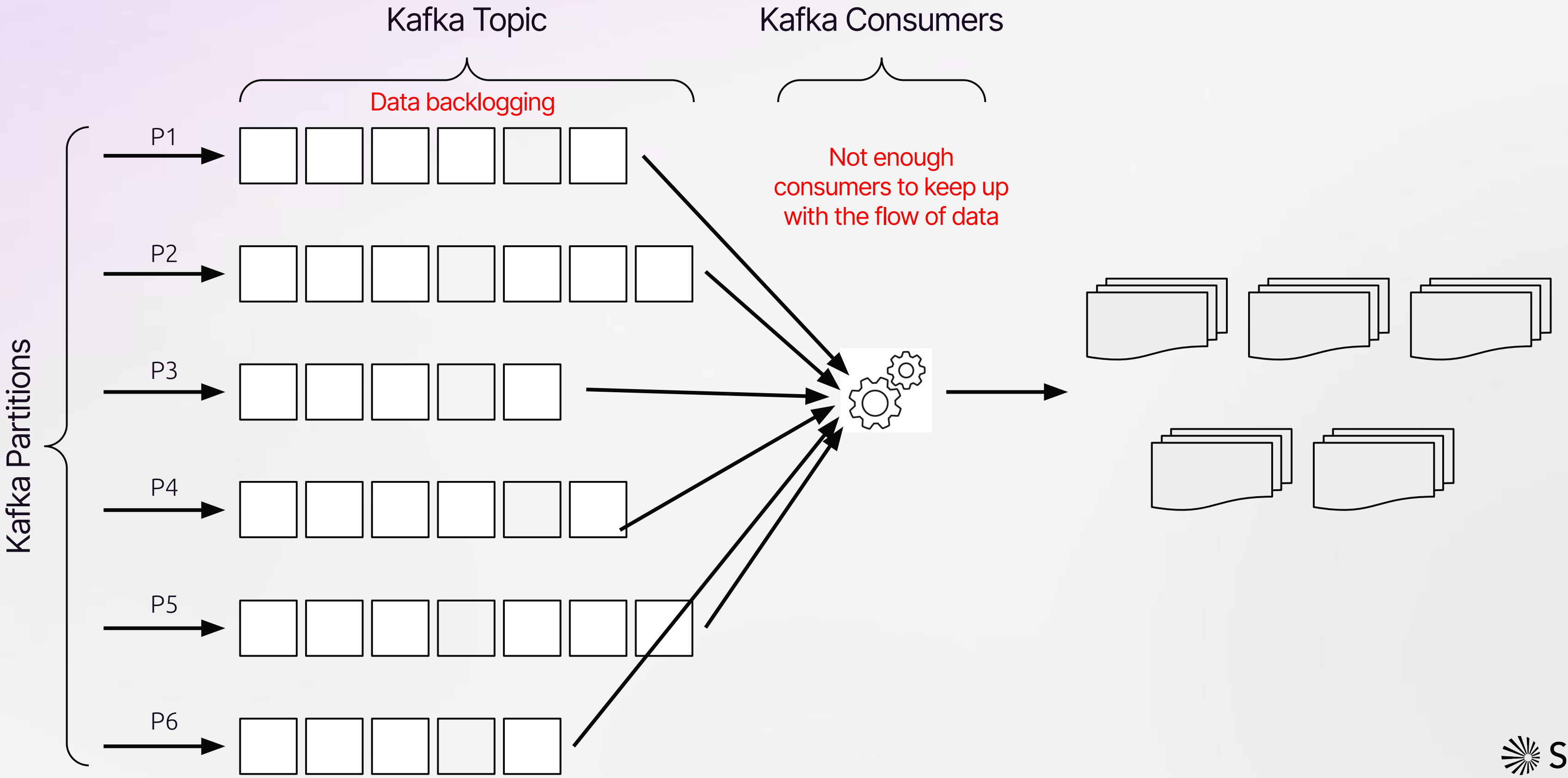


Starburst

Small files kill downstream query performance

Starburst

Challenge - parallelism vs compaction

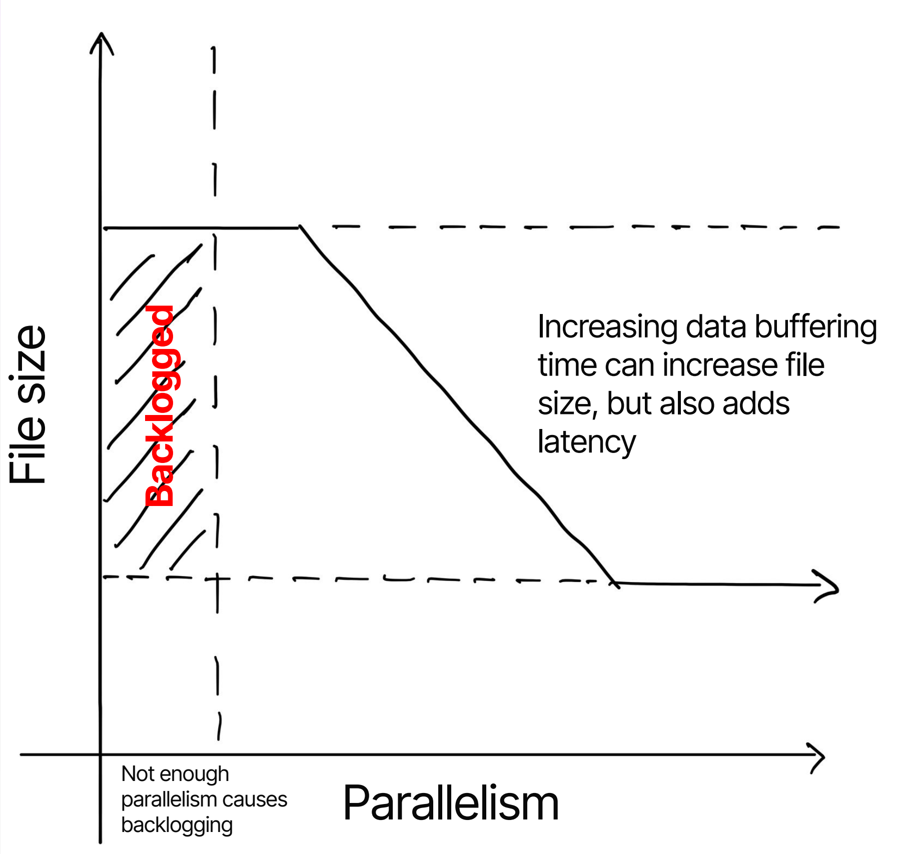


Starburst

Too few consumers causes backlog

Starburst

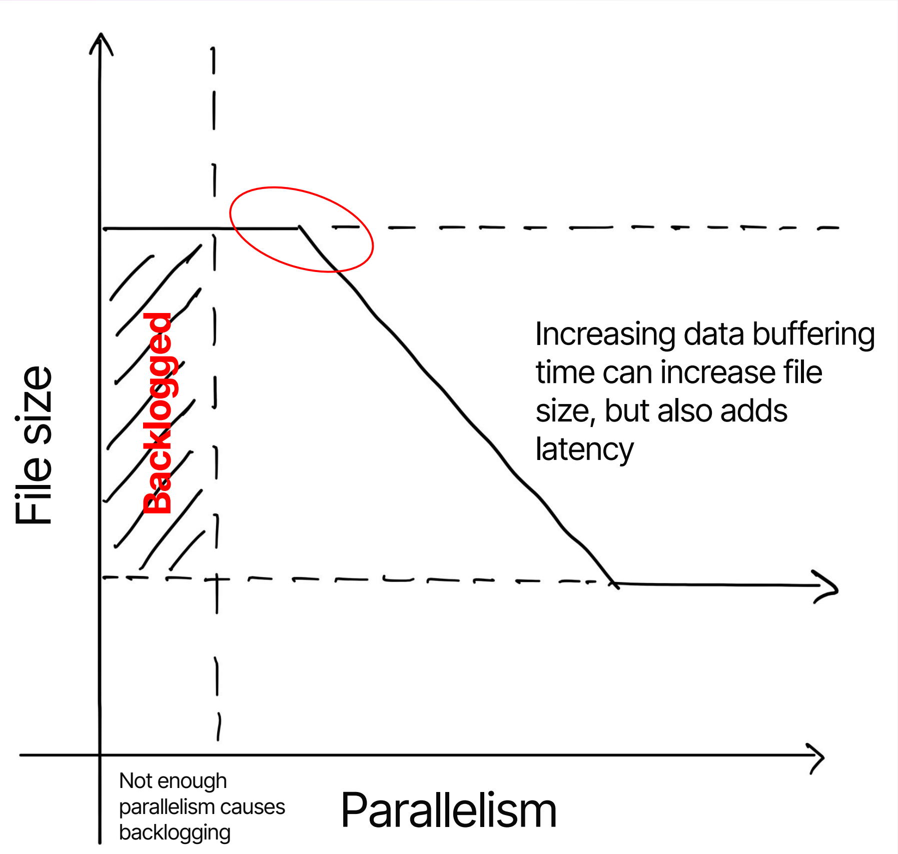
Challenge - parallelism vs compaction



Files fully compacted on write

Kafka partitions fully parallelized

Challenge - parallelism vs compaction



Increasing data buffering time can increase file size, but also adds latency

Not enough parallelism causes backlogging

Files fully compacted on write

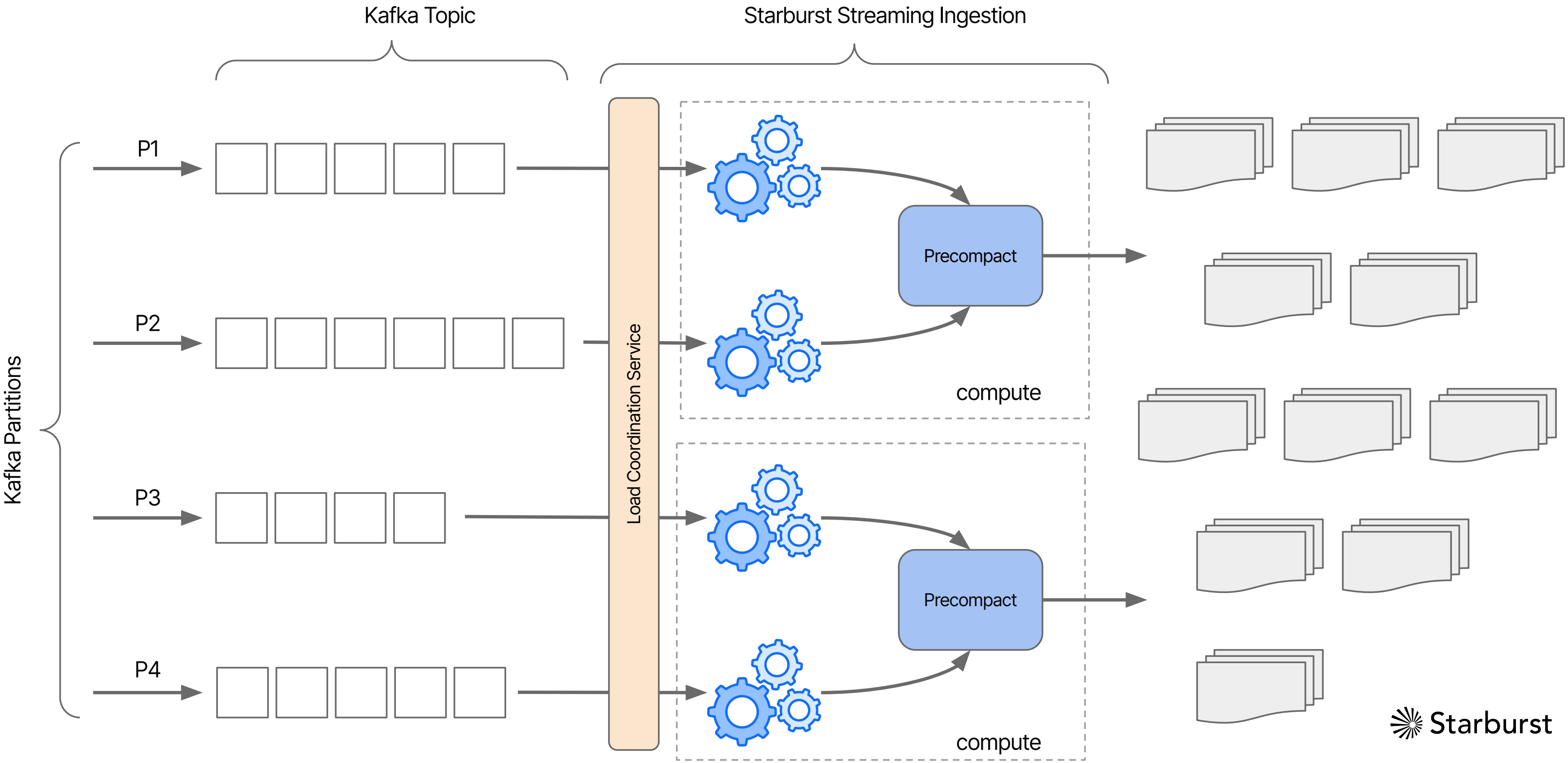
Kafka partitions fully parallelized

How did we solve this?

Dynamic load coordination

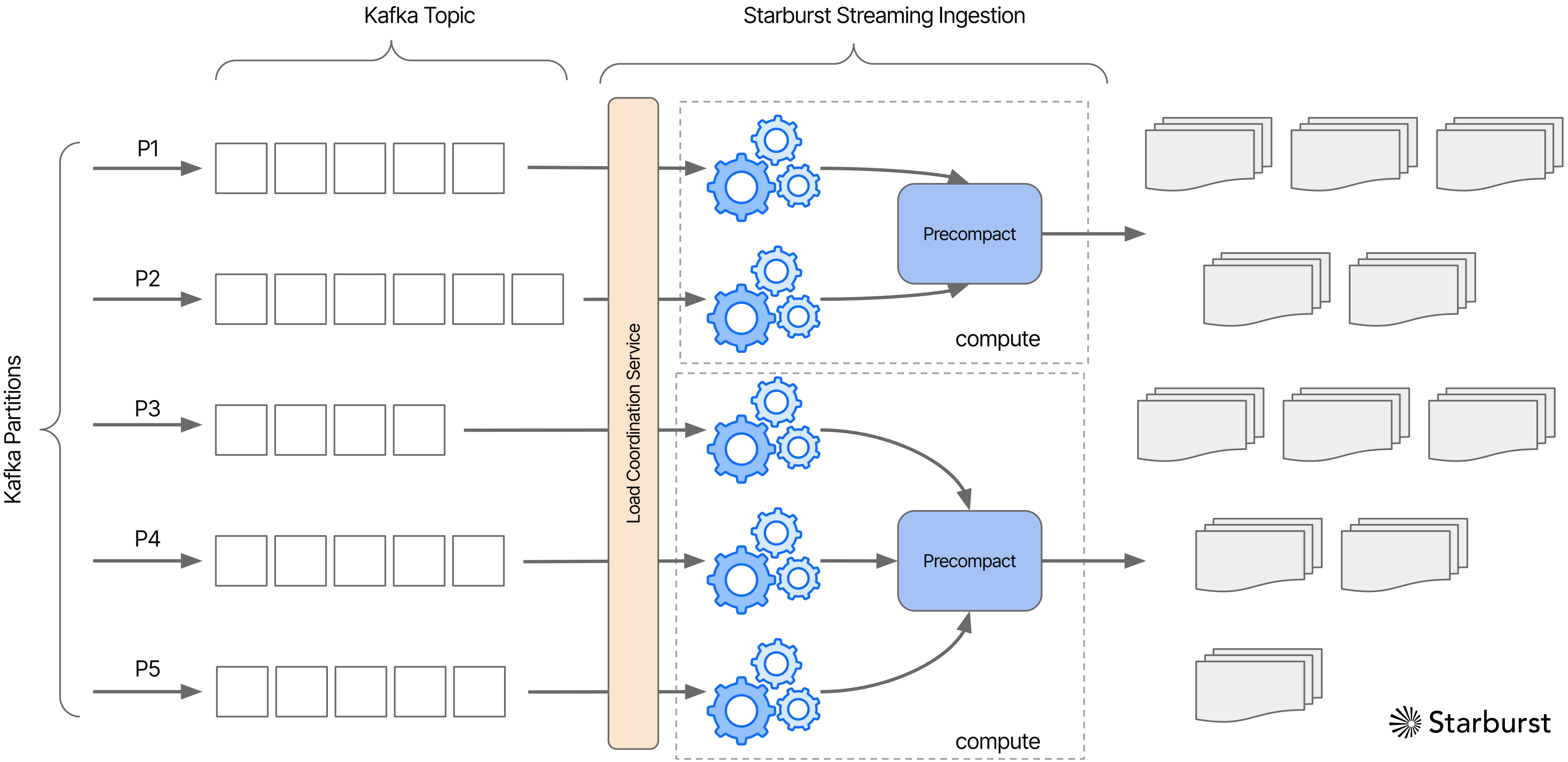
Starburst

Starburst Solution - Dynamic load coordination



Starburst

Starburst Solution - Dynamic load coordination

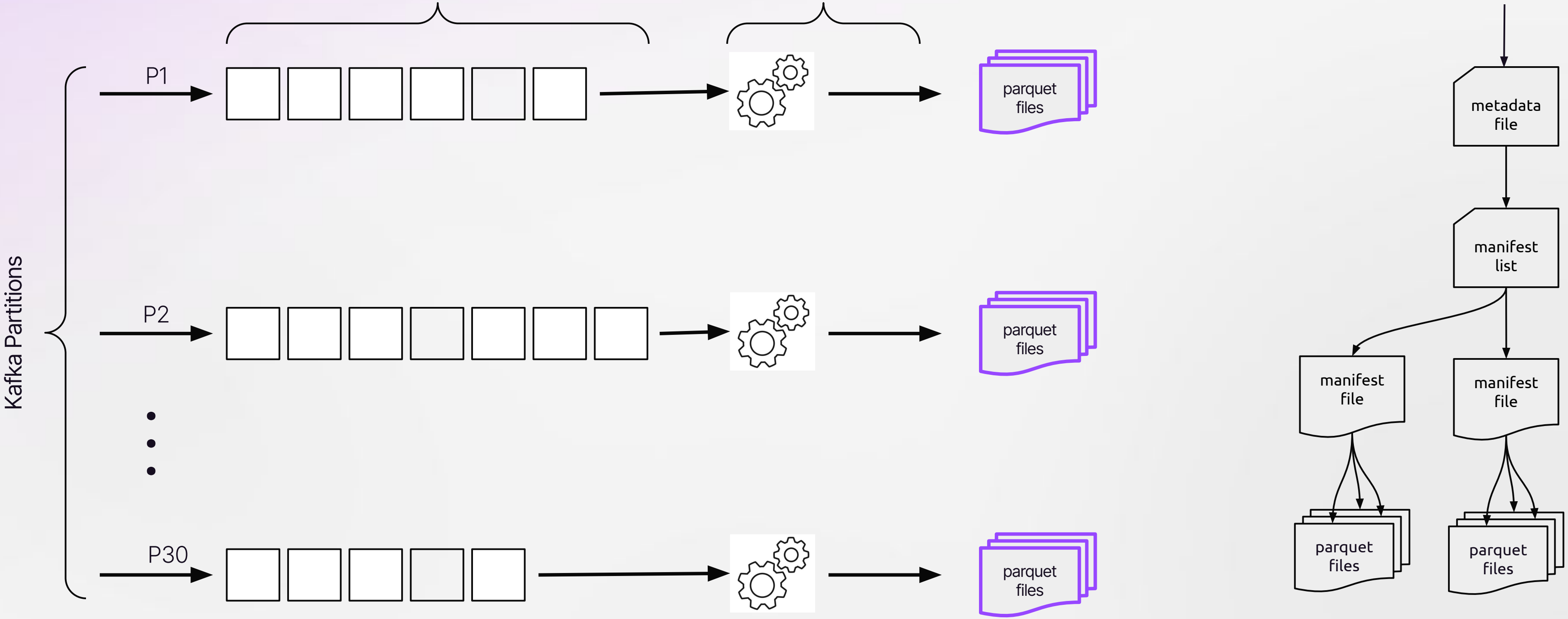


Starburst

Optimizes query performance on near real-time data

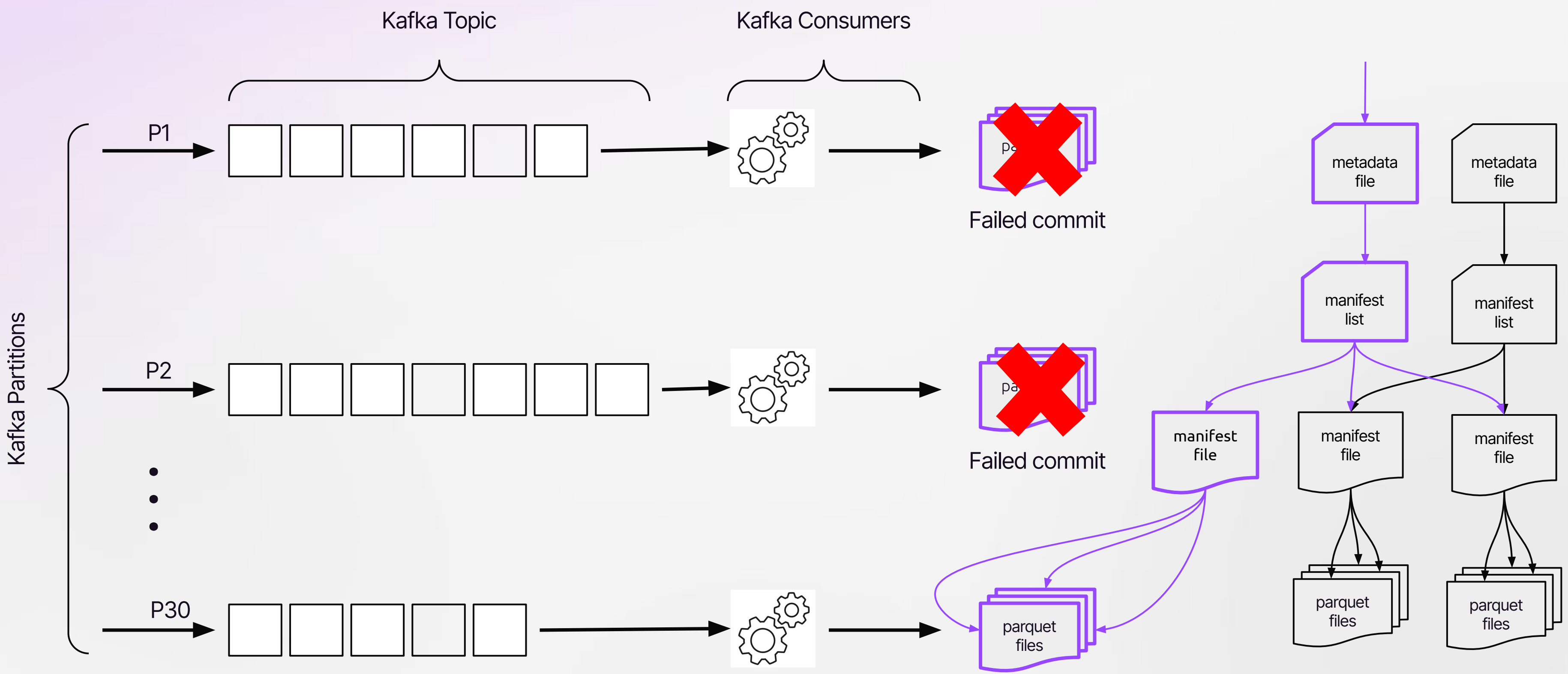
Then there’s Iceberg commit contention problem

Challenge - commit contention



Starburst

Challenge - commit contention



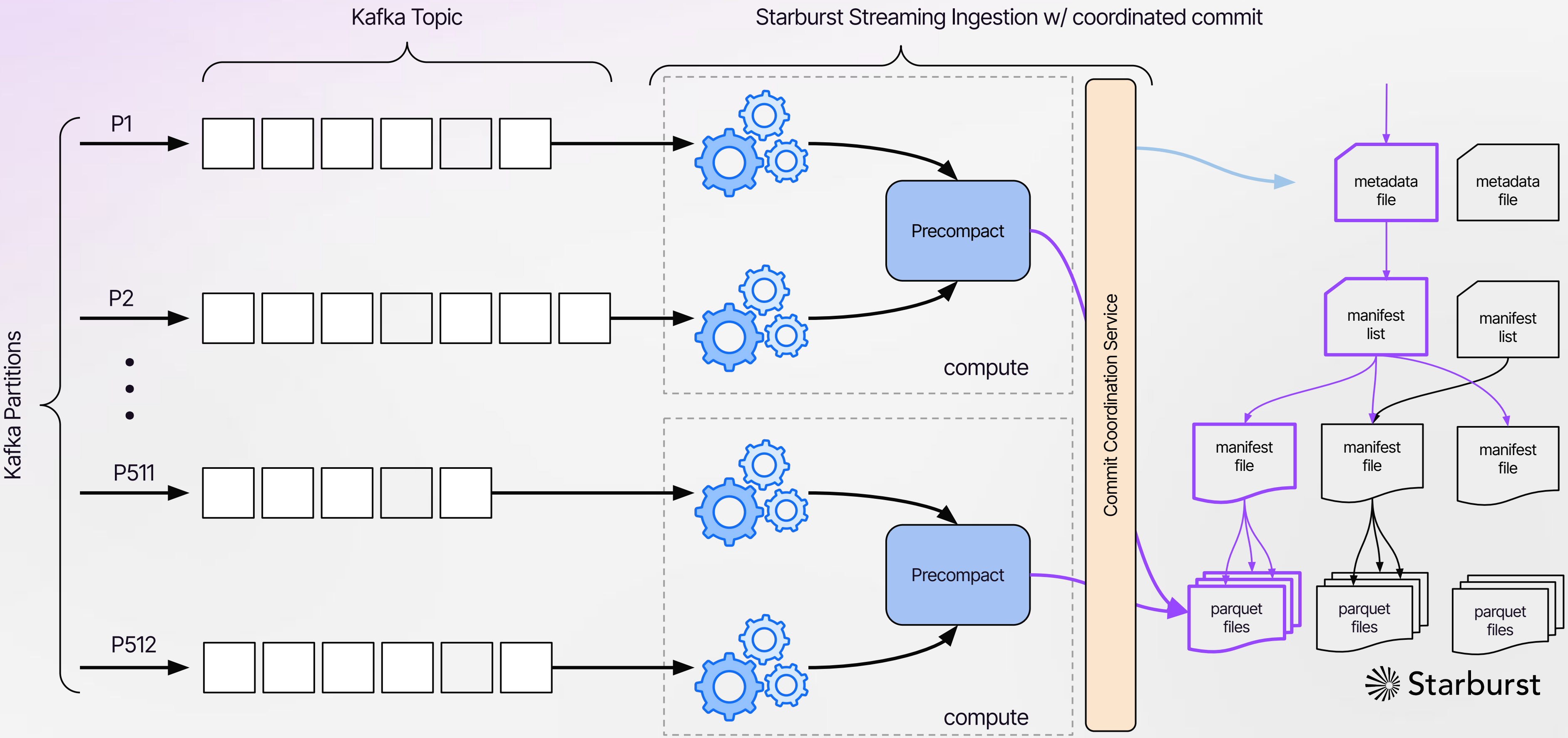
Starburst

How did we solve this?

Iceberg commit coordination

Starburst

Starburst Solution - Iceberg commit coordination



Starburst

Proven internet scale throughput

Tested up to 100 GB per second being written to a single Iceberg table.

Starburst

Proven internet scale throughput

Tested up to 100 GB per second being written to a single Iceberg table.

~1 Petabyte every 3 hours

Unlocking Iceberg streaming ingestion

Building for Scale

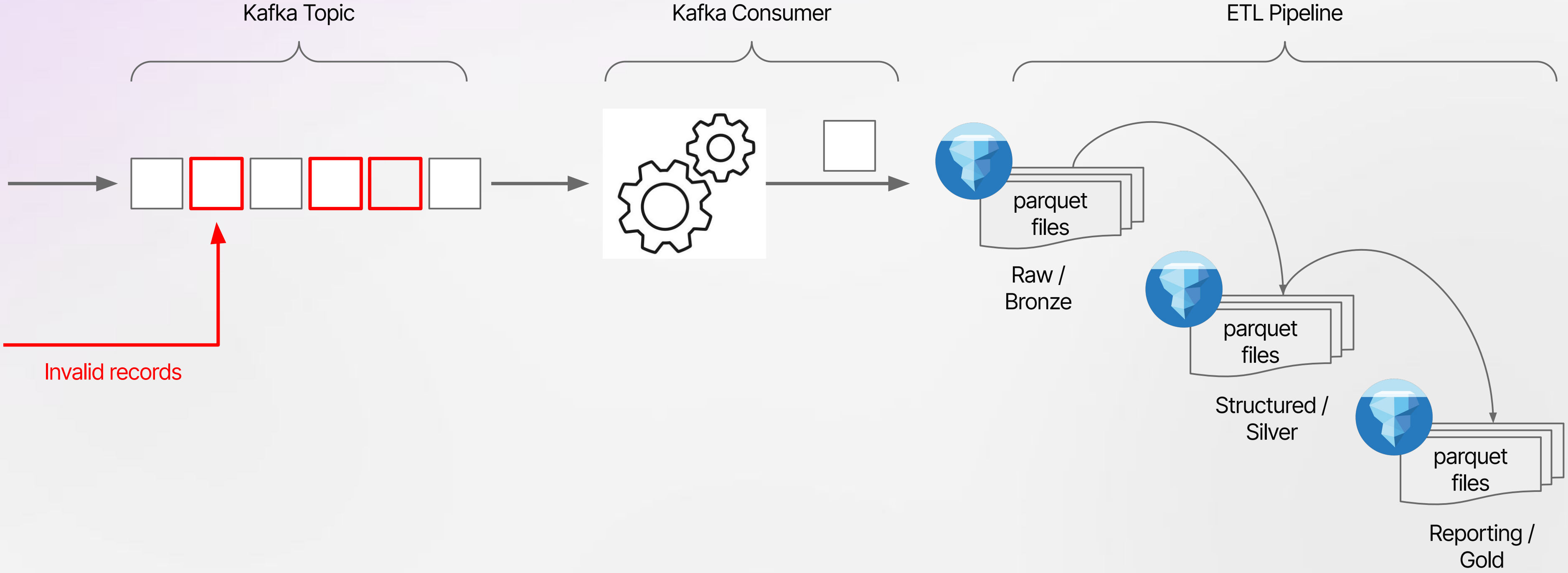
Planning for Imperfection

Unified Data Platform

When things go wrong *(and they will)*

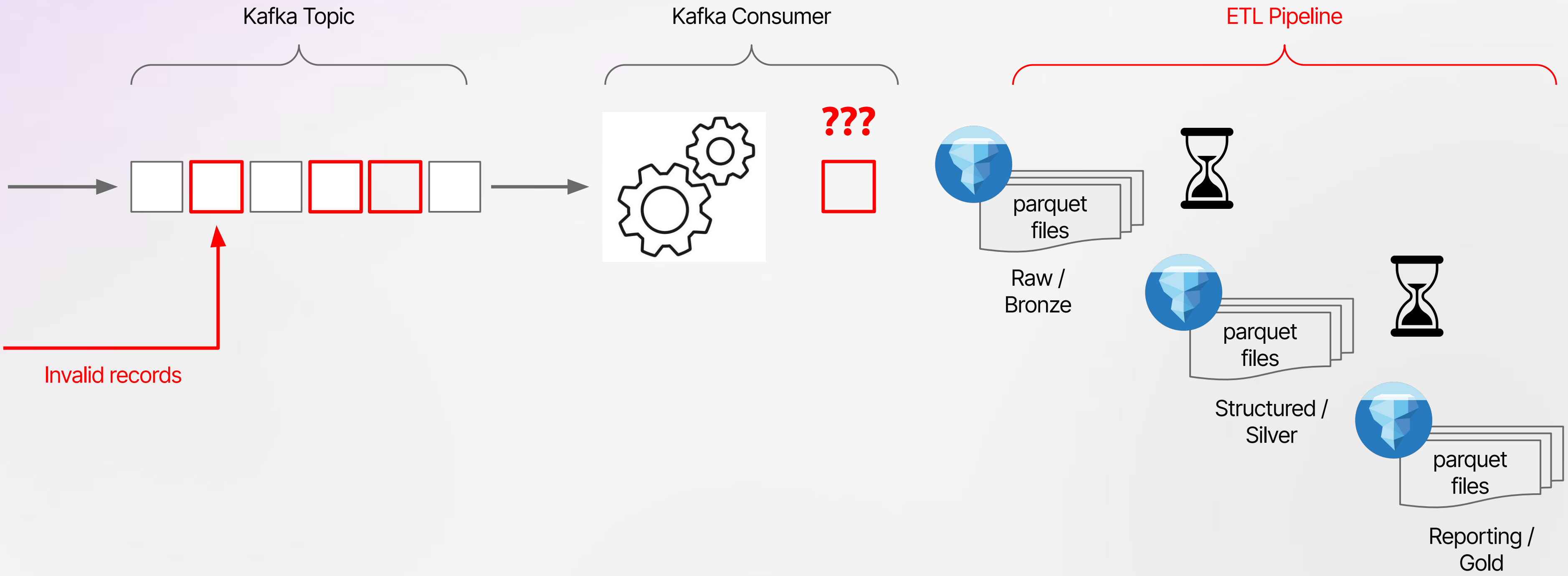
Invalid data

Challenge - dealing with invalid data



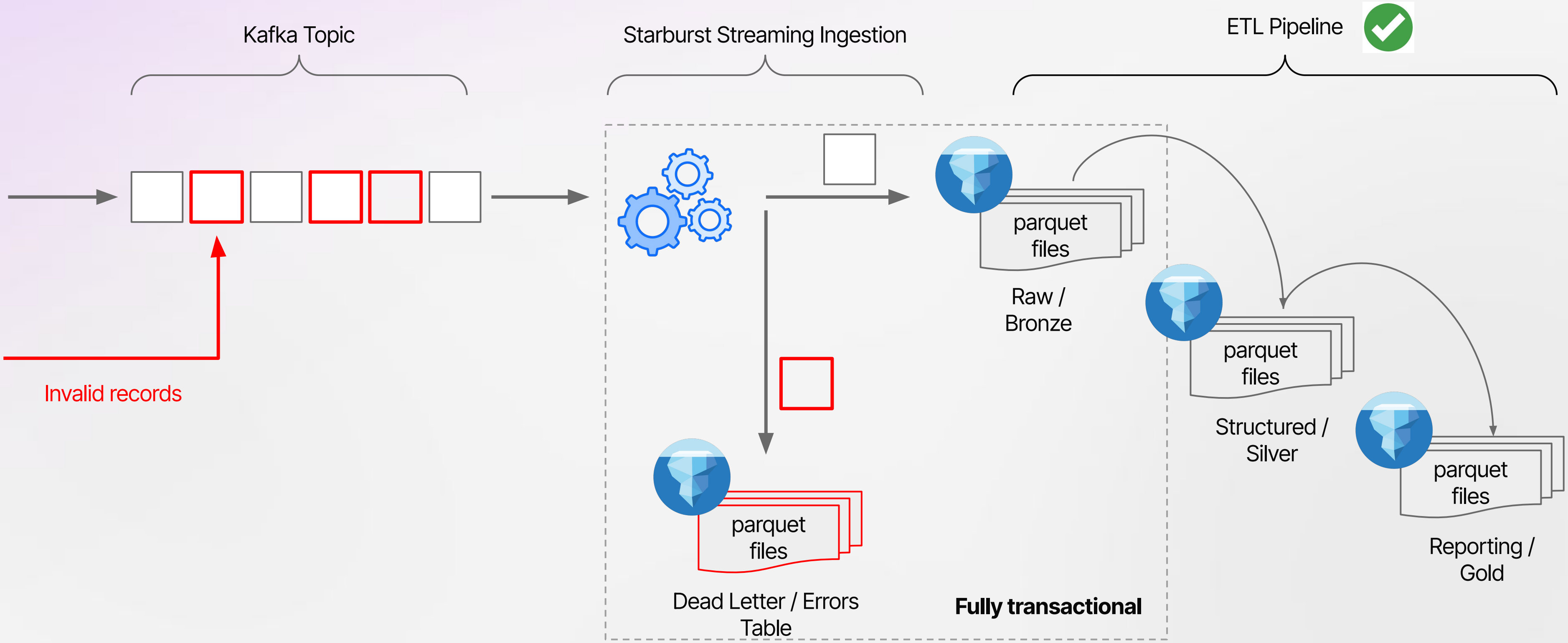
Starburst

Challenge - dealing with invalid data



Starburst

Starburst solution - transactional dead lettering



Starburst

Misconfigured Schema

Starburst

Challenge - Misconfigured schema

Configured Schema

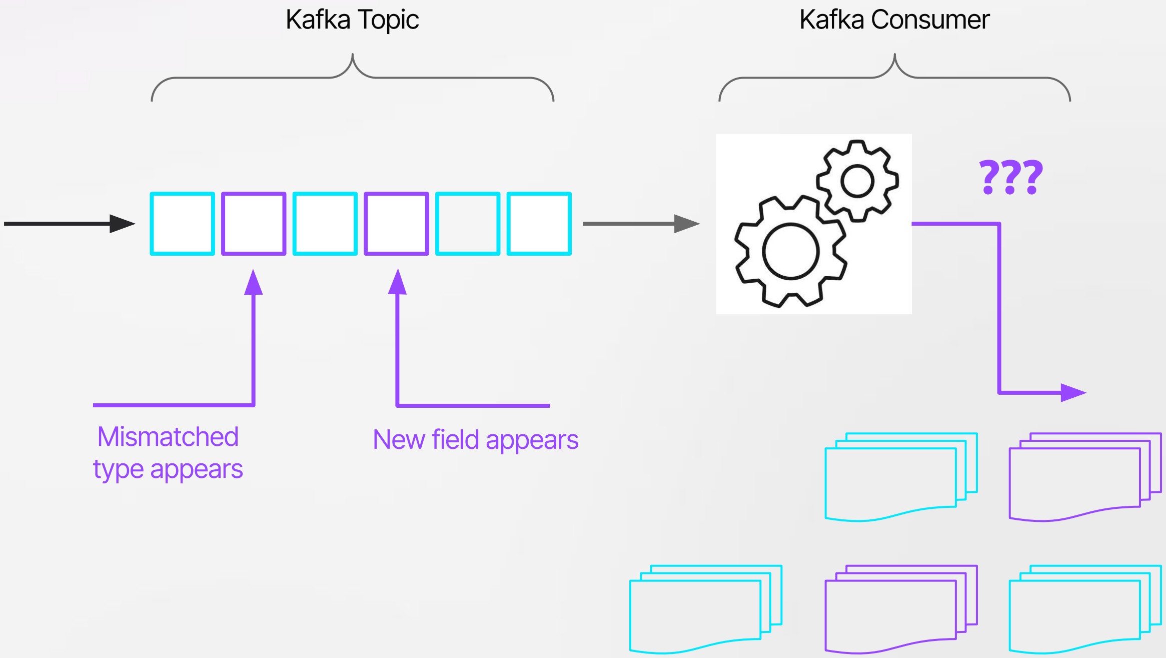
```
"side" -> VARCHAR
"symbol" -> VARCHAR
"quantity" -> INTEGER
"price" -> BIGINT
"account" -> BIGINT
```

Actual message

```
{
  "side": "SELL",
  "symbol": "SNOW",
  "quantity": 1453,
  "price": 25.10,
  "account": 4567,
  "stop_loss": 26.00
}
```

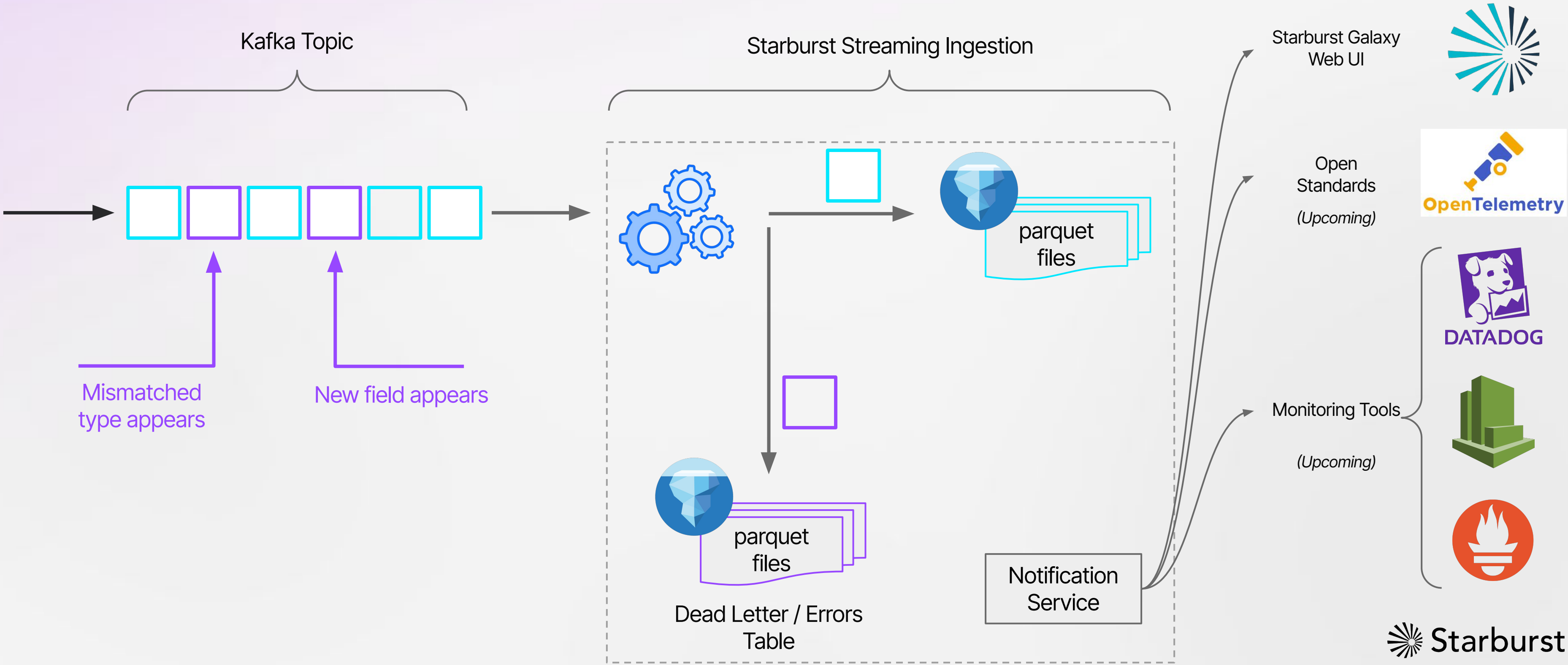
Wrong type

New column omitted



Starburst

Starburst Solution - Error detection, classification, and notification



Starburst

Starburst solution - reset and replay

1. Adjust Schema

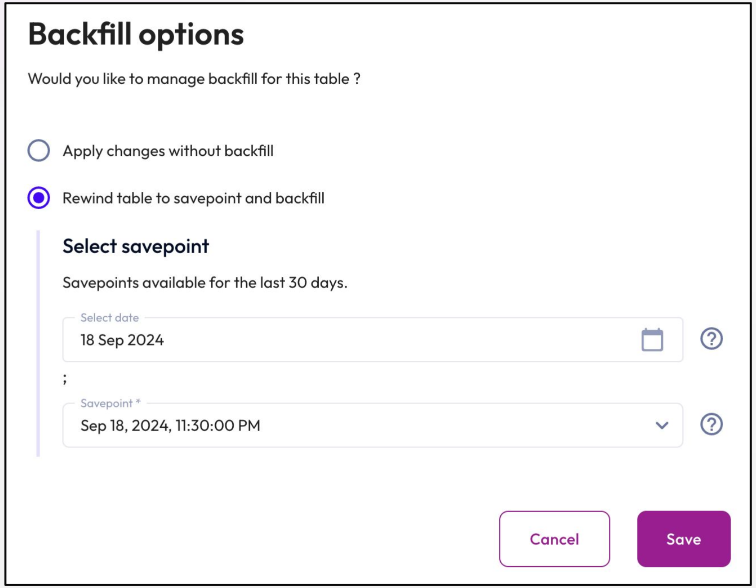
Original Schema

```
"side" -> VARCHAR
"symbol" -> VARCHAR
"quantity" -> INTEGER
"price" -> BIGINT
"account" -> BIGINT
```

New Schema

```
"side" -> VARCHAR
"symbol" -> VARCHAR
"quantity" -> INTEGER
"price" -> DOUBLE
"account" -> BIGINT
"stop_loss" -> DOUBLE
```

2. Choose Reset Point



Backfill options

Would you like to manage backfill for this table ?

○ Apply changes without backfill

● Rewind table to savepoint and backfill

Select savepoint

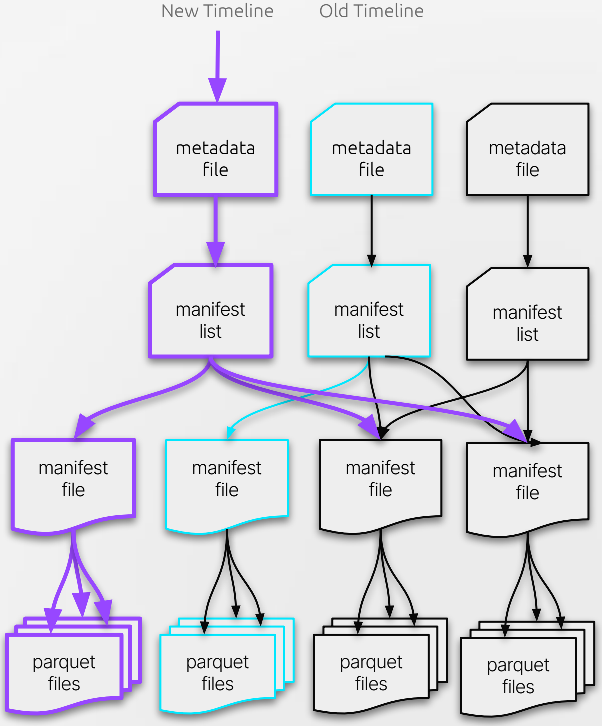
Savepoints available for the last 30 days.

Select date: 18 Sep 2024

;

Savepoint *: Sep 18, 2024, 11:30:00 PM

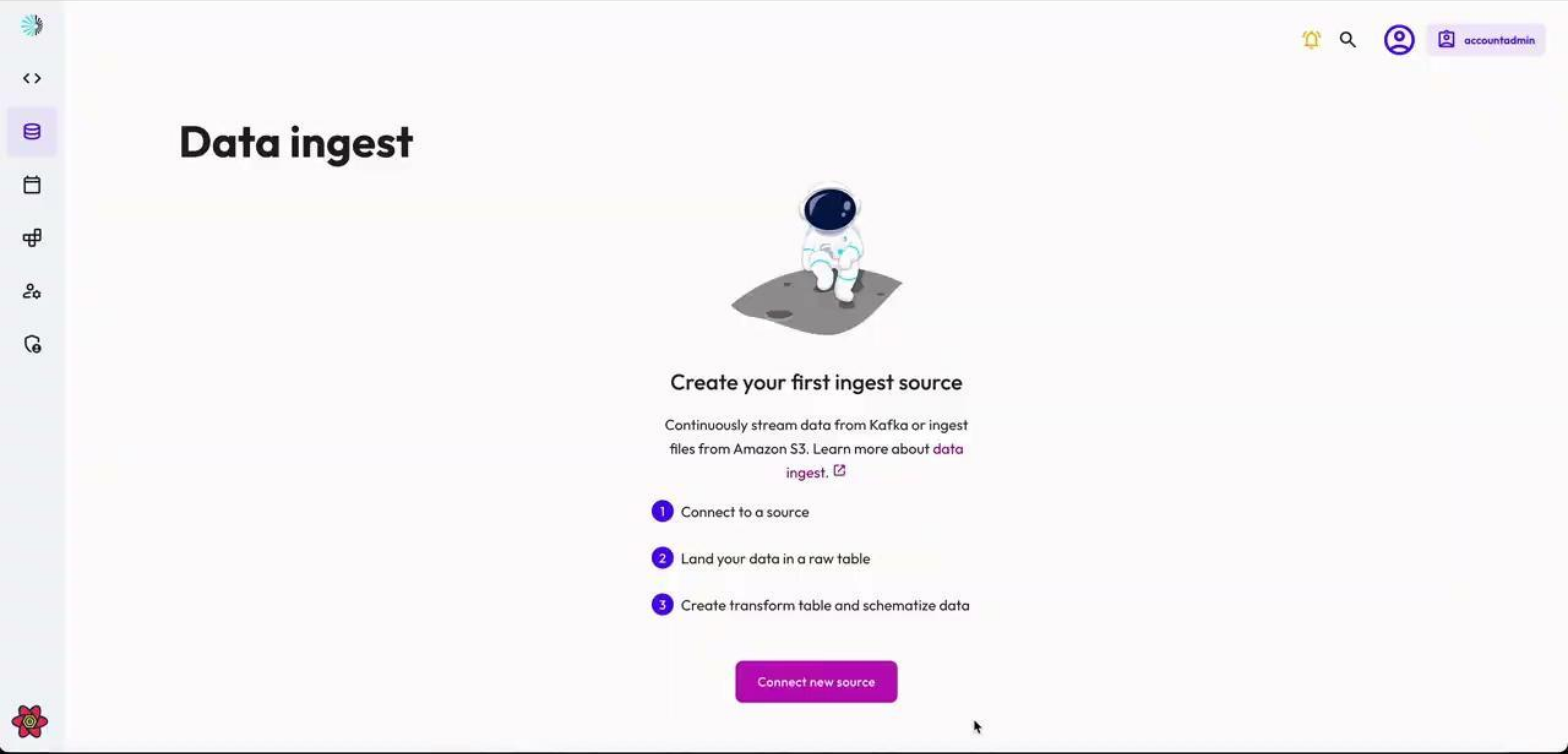
3. Automatically Replay Data Ingestion with New Schema



Starburst

Demo

Starburst



Data ingest

Create your first ingest source

Continuously stream data from Kafka or ingest files from Amazon S3. Learn more about data ingest.

1. Connect to a source
2. Land your data in a raw table
3. Create transform table and schematize data

Unlocking Iceberg streaming ingestion

Building for Scale

Planning for Imperfection

Unified Data Platform

Starburst

Disparate tools vs. Unified platform

- Integration: seems easy, but headache to debug
- Configure and maintain multiple tools

Small frictions, but they add up

- One product for ingest, transform, and analytics
- Faster feedback loop from raw data to decisions
- Simplified operations, stronger governance, and happier teams

Focus on insights, not integration

Leading price performance

Benchmark cost study of 8mb/s throughput (*us-east-1*)

Kafka Ingestion Solution	Monthly Cost	Annual Cost
Starburst Streaming Ingest	$47.16	$565.92
Vendor 1	$79.20 (1.68x)	$950.40
Vendor 2	$224.01 (4.75x)	$2,688.12
Vendor 3	$531.23 (11.26x)	$6,374.76

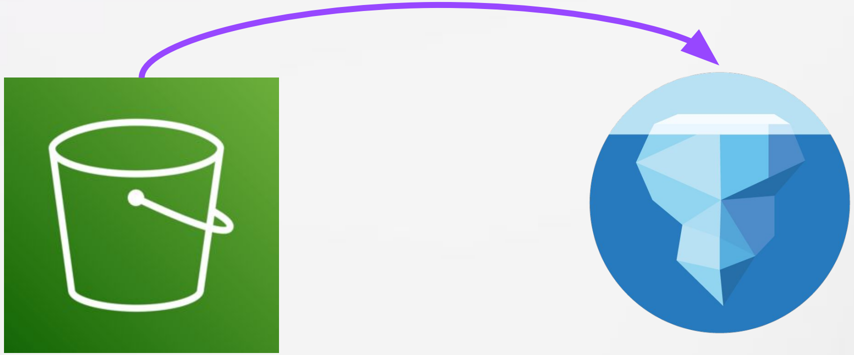
https://www.starburst.io/pricing/pricing-details/

What about other ingestion sources?

Starburst

Iceberg file loader

Continuous S3 to Iceberg ingestion along similar principles



Raw files on
AWS S3

Iceberg Tables on
AWS S3

Starburst Galaxy Ingestion - Summary

Building for Scale

- Dynamic load coordination optimizes query performance
- Commit coordination enables massive scale

Plan for Imperfection

- Transactional dead lettering
- Error detection, classification, notification
- Reset and replay

Unified Data Platform

- End-to-end: from ingest to analytics
- Custom architecture yields price performance advantages

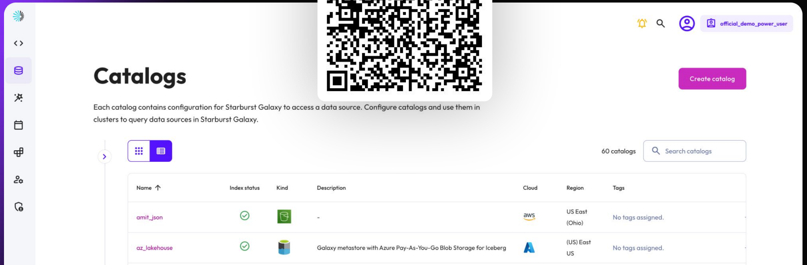
Sign up for our newsletter

and enter to win a limited-edition
Starburst community T-shirt

Join your local Meetup Group

Try Galaxy Free

$500 CREDIT



Catalogs

Each catalog contains configuration for Starburst Galaxy to access a data source. Configure catalogs and use them in clusters to query data sources in Starburst Galaxy.

Name ↑	Index status	Kind	Description	Cloud	Region	Tags
amit_json	✓		-	aws	US East (Ohio)	No tags assigned.
az_lakehouse	✓		Galaxy metastore with Azure Pay-As-You-Go Blob Storage for Iceberg		(US) East US	No tags assigned.