Business insights

Finally, we can use DB SQL queries to create rich visualizations on top of our query results. These visualizations can then be packaged and served to end users through interactive dashboards (see Figure 13).

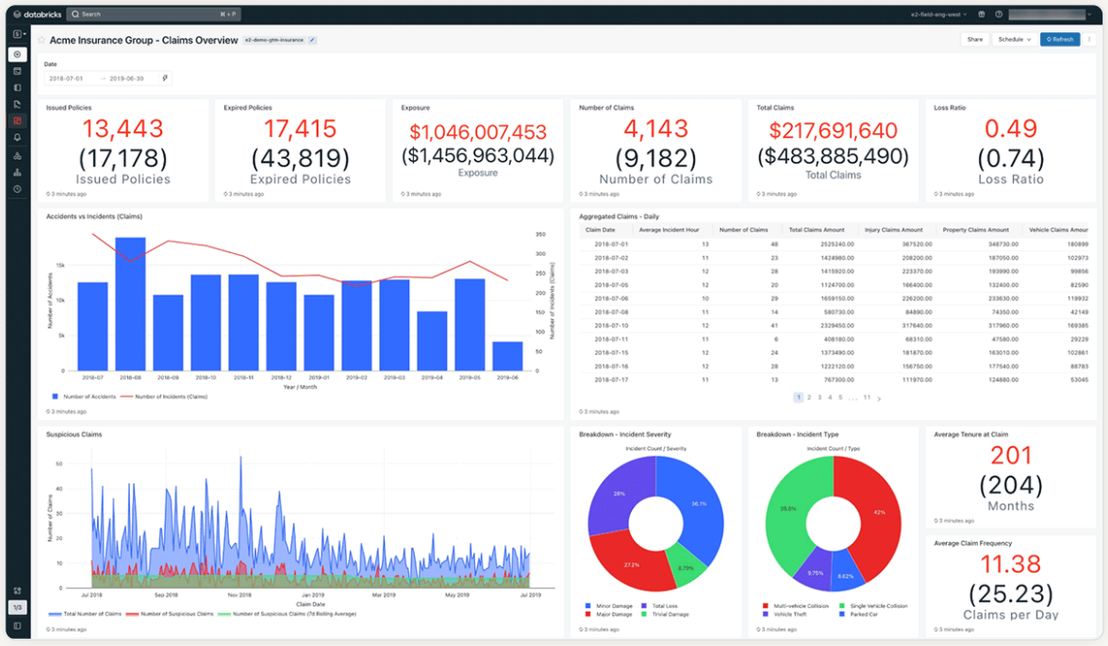

Figure 13 — Example operational dashboard built on a set of resulting Delta Live Tables (DLT) table entities.

For our use case, we created a dashboard with a collection of key metrics, rolling calculations, high-level breakdowns, and aggregate views. The dashboard provides a complete summary of our claims process at a glance. We also added the option to specify specific date ranges. DB SQL supports a range of query parameters that can substitute values into a query at runtime. These query parameters can be defined at the dashboard level to ensure all related queries are updated accordingly.

DB SQL integrates with numerous third-party analytical and BI tools like Power BI, Tableau and Looker. Like we did for Fivetran, we can use Partner Connect to link our external platform with DB SQL. This allows analysts to build and serve dashboards in the platforms that the business prefers without sacrificing the performance of DB SQL and the Databricks Data Intelligence Platform.

Conclusion

As we move into this fast-paced, volatile modern world of finance, batch processing remains a vital part of the modern data stack, able to hold its own against the features and benefits of streaming and real-time services. We've seen how we can use the lakehouse architecture for financial services and its ecosystem of partners to architect a simple, scalable, and extensible framework that supports complex batch-processing workloads with a practical example in insurance claims processing. With Delta Live Tables (DLT) and Databricks SQL (DB SQL), we can build a data platform with an architecture that scales infinitely, is easy to extend to address changing requirements, and will withstand the test of time.

To learn more about the sample pipeline described, including the infrastructure setup and configuration used, please refer to this GitHub repository or watch this demo video.