Title: How to Monitor Stripe Data with a Data Observability Tool like Bigeye

Introduction to Stripe

Stripe is a popular online payment processing platform, enabling businesses to accept payments from customers seamlessly and securely. With its developer-friendly APIs, Stripe has become the go-to choice for many e-commerce platforms, SaaS businesses, and mobile applications. This powerful tool allows companies to manage subscriptions, invoices, and other financial transactions with ease.

Why it's important to monitor your Stripe data with a data observability tool

As your business grows, the volume and complexity of your Stripe data will increase. Monitoring your Stripe data ensures that you can identify issues and opportunities related to your financial transactions, allowing you to make informed decisions about your business. By using a data observability tool like Bigeye, you can automatically monitor and analyze your Stripe data, identifying anomalies and trends that may impact your bottom line.

In this blog post, we will walk you through the process of loading your Stripe data into a data warehouse and using Bigeye to monitor and analyze this data effectively.

How to load Stripe data into a data warehouse

The easiest way to load your Stripe data into a data warehouse is to sync Stripe with Snowflake or Amazon Redshift using Stripe's built-in integration. Both Snowflake and Redshift are cloud-based data warehouse solutions that can efficiently store and analyze large volumes of data.

To set up the integration, follow these steps:

1. Sign in to your Stripe account.
2. Navigate to the "Integrations" tab.
3. Search for "Snowflake" or "Amazon Redshift" in the list of available integrations.
4. Follow the on-screen instructions to connect your Stripe account to your chosen data warehouse.

Once the integration is complete, Stripe will automatically sync your data with your chosen data warehouse, making it easy to monitor and analyze your financial transactions.

Example Stripe data

Here's an example of a simple Stripe dataset containing transaction details:

ID AMOUNT AVAILABLE\_ON FEE NET AUTOMATIC\_TRANSFER\_ID

bt\_xcVXgHcBfi83m94 500 4/24/2023 50 450 po\_rC4ocAkjGy8zl3j

This dataset contains information such as the transaction ID, amount, available date, fees, net amount, and automatic transfer ID.

Main metrics that should be tracked over Stripe data

When monitoring Stripe data, it's essential to track the following key metrics:

1. Gross Revenue: The total amount of money generated from all transactions, including fees.
2. Net Revenue: The total amount of money generated from all transactions, excluding fees.
3. Fees: The total amount of fees charged by Stripe for each transaction.
4. Refunds: The total amount of money refunded to customers for canceled or disputed transactions.
5. Chargebacks: The total number of chargebacks filed by customers due to disputes or fraudulent charges.
6. Disputes: The total number of disputes raised by customers or financial institutions.
7. Failed Payments: The total number of failed payment attempts, indicating potential issues with your payment processing setup.

Using Bigeye to monitor your Stripe data

Bigeye is a data observability platform that allows you to monitor and analyze your data in real-time. To use Bigeye to monitor your Stripe data, follow these steps:

1. Sign up for a Bigeye account if you don't have one already.
2. Navigate to the "Data Sources" tab in the Bigeye dashboard.
3. Click on "Add Data Source" and select "Stripe" from the list of available data sources.
4. Connect your Stripe account to Bigeye by following the on-screen instructions.
5. Once your Stripe data is connected to Bigeye, you can create custom metrics and alerts to monitor

When Stripe data is loaded into Snowflake, it is usually organized into multiple tables, each representing a specific entity or event type within the Stripe ecosystem. Some of the most common tables you'll find in Snowflake when working with Stripe data are:

1. charges: This table contains details about each charge created on Stripe, including the amount, currency, payment method used, customer details, and the outcome of the transaction. It also includes metadata like timestamps and identifiers.
2. customers: This table stores information about customers, such as their name, email, address, and other relevant details. It also contains metadata, like customer IDs and timestamps, to help track the customer's interactions with your business.
3. invoices: The invoices table contains data about invoices generated by Stripe, including the invoice amount, due date, status, and line items. It also includes metadata like invoice IDs and timestamps.
4. refunds: This table stores information about refunds issued to customers, including the refund amount, currency, and the reason for the refund. It also contains metadata like refund IDs and timestamps.
5. subscriptions: The subscriptions table contains data about customer subscriptions to your products or services, including the subscription plan, start and end dates, and the status of the subscription. It also includes metadata like subscription IDs and timestamps.
6. disputes: This table holds information about disputes raised by customers or financial institutions, including the reason for the dispute, the status, and any supporting evidence provided by the customer. It also contains metadata like dispute IDs and timestamps.
7. transfers: The transfers table contains data about transfers made between Stripe accounts or to connected bank accounts, including the transfer amount, currency, status, and destination account. It also includes metadata like transfer IDs and timestamps.

When working with Stripe data in Snowflake, you can join these tables using their respective IDs (e.g., customer\_id, charge\_id, invoice\_id) to gain insights into your financial transactions and customer behavior. Additionally, you can create custom views and queries to analyze the data based on your specific business needs.