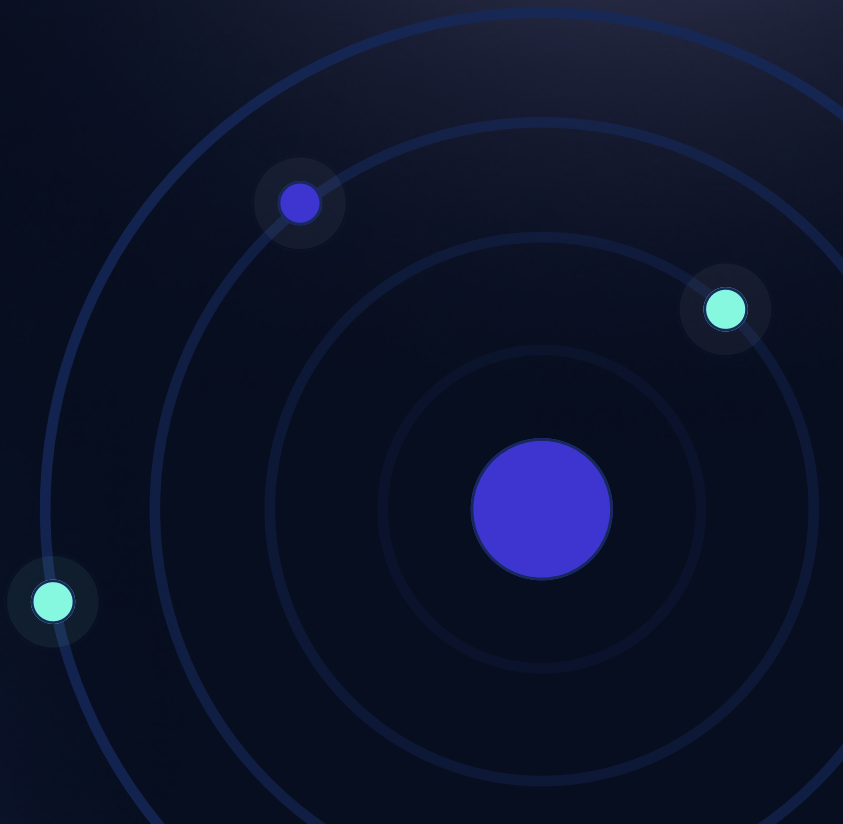
rudderstack™

WHITEPAPER

Guide: Creating a Warehouse-First Data Analytics Stack



Guide: Creating a Warehouse-First Data Analytics Stack

Suhail Doshi, founder of Mixpanel, once said: "Most of the world will make decisions by either guessing or using their gut. They will be either lucky or wrong." For organizations to make intelligent business decisions, like deciding what new business opportunities to pursue or how to reduce customer churn, they need data.

But, according to Gartner's 2020 Analytics Survey, the reality is that, even with data, most companies are still unable to derive meaningful business insights from their analysis. The reason? Very few companies correctly combine disparate sources of data from marketing, sales, product, and finance teams and rarely apply the business context to sampled data. However, adopting a warehouse-first data analytics approach overcomes these issues by centralizing all of your company's data in one location, which allows you to have complete control and access to all your data.

WHAT EXACTLY DOES “WAREHOUSE-FIRST” DATA ANALYTICS MEAN?

A warehouse-first data analytics stack is an analytics stack that has a single data repository (otherwise known as the data warehouse) that all customer data is fed into. Analytics tools (e.g., Google Analytics) and other customer tools (e.g., Salesforce) used within organizations create data silos and cannot communicate with each other. This makes it challenging for organizations to access all the data they need to make complex decisions.

By gathering all of your company's data into a single location, you're able to build efficient analytics on large, diverse, high-quality datasets to answer questions that your analytics tools are not able to answer on their own.

For example, suppose your company is a startup in its early stages. In that case, you're probably just trying to understand how customers perceive or interact with your product, so you likely have questions like:

- How many active users did we have this month?

- What's our overall retention rate?
- What are our top sources of referral signups?

In such cases, using only tools like Google Analytics (GA), Mixpanel, or Amplitude will do. But as your company grows, you'll need answers to more complex questions like:

- What behavior in a user's first day indicates they're likely to sign up for a paid plan?
- What patterns or conditions lead to an increase or decrease in purchases or usage of our products?
- What products should we cross-sell or recommend to specific users to increase revenue?
- How should we price our product based on the cost of acquisition or cost of resources per customer?

The analytics tools mentioned above cannot answer these questions on their own because they don't have access to all the data required. For example, in a retail setting, let's say you want to predict what products individual customers are likely to buy so you can make tailored recommendations. You'll need to consider a variety of data from different sources, such as:

- User events data tracked by your analytics tool
- Transactional or payment history, which may be stored in Stripe or internal systems
- Engagement with previous marketing campaigns, which may be stored in Marketo or Facebook Ads Manager
- Previous complaints or communications with your organization that may be stored in a CRM tool like Salesforce or Zendesk, as well as other customer tools used within your organization

But, because all these different types of data are siloed in individual platforms that your analytics tools don’t not have access to, it becomes a difficult or impossible task.

In a previous article, we discussed the benefits of adopting a warehouse-first approach to analytics. In this article, we'll discuss the different tools to consider to put together an optimized warehouse-first data analytics stack — one that allows for valuable analytics to be produced.

THE ELEMENTS OF A WAREHOUSE-FIRST DATA ANALYTICS STACK

A warehouse-first data analytics stack is made up of a few crucial elements. You'll need a few categories of tools that facilitate the collection, storage, and movement of the data to make it easier for you to analyze.

For each element, let's look at a few tools fit to get the job done.

A place where all the data goes: data warehouses

Having all of your company data in a single place not only gives you complete ownership over all of your data, but you can also perform more profound analyses. It also means that you can easily move to a different analytics tool at any time because you can import your historical data.

Numerous data warehouse options are available, Redshift, Snowflake, and BigQuery are the most popular. But, before you choose a data warehouse, you need to consider factors like:

- **What type of data you will be storing:** Will it be structured data (data that fits well into rows and columns) or semi-structured data (such as emails, social media posts, or geographical data)?
- **The quantity of data you plan to store in the data warehouse:** For most use cases, there's no need to worry about this because any data warehouse is typically able to store massive amounts of data without additional costs.
- **How easily can the data warehouse scale:** Are there enough storage and compute resources to process data in times of demand or peak season without affecting performance?
- **Management requirements:** Are you willing to dedicate engineering time to vacuuming, resizing, and monitoring the cluster to ensure performance remains strong? For smaller teams, it would make sense to have a fully managed, self-optimized data warehouse, so your engineers can focus on building your products. However, manually managing a warehouse means you have more flexibility and control and can optimize it precisely for your company's needs.
- **How much it costs:** A data warehouse’s pricing structure is based on a mix of storage capacity, run time, and queries. If you frequently execute queries on your data, you should opt for a solution with a reduced compute cost.

Redshift

Redshift is a data warehouse owned by Amazon Web Services (AWS). Redshift is a relational data warehouse and, therefore, accepts only structured data types. Redshift requires some sort of management in the sense that, in times of high demand, if you need to scale, then you need to handle that manually by adding new nodes. Usage costs a minimum of $0.25 based on the type and number of nodes in your cluster. In general, Redshift may be an ideal choice for organizations that have already invested in AWS tooling and deployment for seamless integrations with other AWS offerings.

Snowflake

The Snowflake data warehouse has a columnar database engine capability, which means that it can handle both structured and semi-structured data, such as JSON and XML. It automatically scales up or down depending on demand and is fully managed with automated administration and maintenance. In addition, it has a decoupled architecture that allows for computing and storage to scale separately, with data storage provided on the user's cloud provider of choice (Google Cloud, AWS, or Azure). Snowflake's pricing is based on the volume of data you store in Snowflake and the compute time you use. So, you can turn off compute resources when they're not in use. If you've already been making use of any of the three cloud storages mentioned and wouldn't want to switch, then snowflake may be an ideal choice for you.

BigQuery

BigQuery is a serverless, highly scalable, and cost-effective multi-cloud data warehouse built by Google Cloud. It is based on Google's internal column-based data processing technology, "Dremel", and therefore is able to handle both structured and semi-structured data. It is also a fully managed data warehouse and can automatically allocate computing and storage resources as you need them.

BigQuery has two pricing models: Pay per number of bytes processed per query or pay a flat rate by purchasing dedicated virtual CPUs for a certain period of time. One key differentiator of BigQuery is its integration with BigQuery ML. So, if you're looking to build machine learning models on top of your data warehouse to aid predictions, that may be a good reason to go for BigQuery.

Tools that transport data: data pipelines

Once you've ticked off securing a data warehouse, data pipelines are the next thing to think of. In simple language, a data pipeline can be thought of as a data transporter. It gets data from a source, performs some data processing actions on the data, and then transports it to a destination.

In a warehouse-first data analysis setup, three types of data movements are required:

- Move user event streams from the app or website to a warehouse and other analytic or customer tools that need it.
- Move non-user events trapped in customer tools like Zendesk or Salesforce into the warehouse.
- Move data models from the warehouse and into the different customer tools for analysis activation.

The first type of movement is called event streaming. To put things into perspective, let's assume that the marketing and product teams in your organization want to track the event of user signups on your website or mobile app. The marketing team wants this data in Google Analytics to track conversions around ad spending and other campaigns. Meanwhile, the product team wants the data in Amplitude to track user journeys and user flows.

Normally, SDKs from both tools will be added to the website or app source code to track this event. This will entail two separate HTTP calls and network calls, which will add extra performance overhead. Imagine the overhead if there were five such tools! Instead, with an event streaming data pipeline tool, you can instrument your website and app with just one SDK. The event streaming tool will consume the event, transform it into the different format that the two tools will accept, and send it to them and the data warehouse. No more dealing with API changes and broken pipelines.

This approach also helps when a team wants to switch tools and add a new tool to the stack. You won't need to touch the codebase to delete or add any new SDKs. Instead, you can simply remove the tool or integrate the new tool via the event streaming tool's dashboard. Some companies build in-house solutions with tools like Apache Kafka or Amazon Kinesis, but most companies opt for systems that are easy to implement and manage, like Segment and RudderStack.

The second type of data movement is moving non-user events data siloed in different cloud customer tools, such as Facebook Ads Manager or Marketo, into your warehouse. This is

important because event data captured from your digital products (websites or apps) is only a subset of customer data and does not offer a detailed, unified view of customer identities on its own.

The tools in this category are called ELT tools. ELT stands for "extract, load, transform," which are the steps required to take your data from a system like Salesforce or QuickBooks and insert it into your data warehouse. Tools in this category are Fivetran and Stitch.

The third type of movement is from the data warehouse to other cloud tools. After you've modeled the data on your warehouse (more on this in the next section) into data models, how do you get the data models out of the warehouse and into the different analysis activation tools like Marketo, Kissmetrics, or Optimizely? This process is known as reverse ELT. Examples of tools in this category are Census and Hightouch

As modern companies move towards a warehouse-first architecture, many are finding that managing multiple vendors for these three separate pipelines is both painful and complex. Enter RudderStack.

RudderStack can be thought of as an open-source combination of Segment + Fivetran + Hightouch. It's an all-in-one customer data pipeline. It can capture event data from your digital products and send it to your data warehouse and other tools with its Event Stream feature. It uses the Cloud Extract feature to aggregate and correlate non-event data with event data in the data warehouse. And finally, to get data out of the data warehouse, it has a reverse ELT feature known as Warehouse Actions. Also, its source code is publicly available on GitHub, so you can choose to self-host or use RudderStack Cloud for a fee (but you can get started for free).

Getting the data ready for analysis: data modeling tools

When the data pipeline has done the job of pulling in encompassing customer data from different sources and events into the data warehouse, the next step is to make sense of all that data.

Data modeling is all about organizing, transforming, and grouping the data to satisfy a particular purpose. For example, say you need to generate a report of the highest lifetime value customers within the company. You will need to organize all the required data scattered over the data warehouse into one dataset from which the report can be generated.

Once the data is modeled, it is much easier to extract value from it, whether in the form of dashboards or reports or as a base for predictive or prescriptive analytics.

When choosing a data modeling tool, it's important to go with one that allows you to model your data right within the warehouse instead of exporting it to a different tool.

dbt

One of the most popular data modeling tools is dbt built by dbtlabs. dbt (data build tool) enables data analytics engineers to transform data in their warehouses by simply writing SQL select statements. dbt takes those SQL codes and runs them against your data warehouse to create tables and views.

dbt enables data engineers to work like software engineers with version control, continuous integration, and testing built-in. Because dbt is SQL-based, it is straightforward to get started with. Plus, dbt is open-source and has a very active community on Slack.

LookML

Another tool in the data modeling category is LookML. LookML is embedded as a modeling layer/component into the Looker business intelligence/visualization tool by Google Cloud. LookML bakes in some neat data types that aren't necessarily baked into SQL or your database. However, to use LookML, you need to move data from your warehouse into Looker. This means that the data models you create will be locked in Looker, and you can't send it out to other tools.

For example, here at RudderStack, we use dbt to model and create datasets. The datasets are then stored as tables in the warehouse, which can be sent to Looker or other platforms like Salesforce or Mixpanel — whenever we wish.

Making use of the data: visualization tools

Data modeling is not complete without visualization. Now that you've transformed the data in the warehouse into rich datasets, the next step is to feed it into a visualization tool that will allow you to visually represent those datasets in the form of charts, maps, graphs, or images in order to draw valuable insights from them.

There are a lot of visualization tools in the market. Some of the popular ones are Tableau, FusionCharts, and Metabase. However, before you pick a tool, you should consider things like ease of use, learning curve, support for your choice of data warehouse, level of flexibility, customization options, and if it fits your use case.

Tableau

Tableau is often recognized as the undisputed king of data visualization software and with good cause. Due to its ease of use and capacity to generate interactive visuals well beyond standard BI solutions. It is especially well suited to dealing with large and rapidly changing datasets.

Metabase

Metabase is an open-source business intelligence tool that lets you create charts and dashboards using datasets from your data warehouse. Although SQL is not required to produce visualizations, Metabase does allow SQL for sophisticated customization. Its simplicity and ease of use are the top reasons why many users love it.

FusionCharts

FusionCharts is a popular JavaScript-based visualization tool. One feature contributing to FusionCharts' popularity is that instead of starting each new visualization from scratch, you can use various "live" example templates by simply putting in your datasets.

Expanding the stack: warehouse machine learning tools

Your analysis does not have to end at modeling and visualization. The beauty of the warehouse-first data analytics stack is that with all the vast amounts of data in your warehouse, you have the kind of data needed to build machine learning models. You can go a step further to generate inferences in the form of predictive or prescriptive models for things like recommendation engines, churn prediction, or predictive maintenance of server loads.

This is possible with the help of some tools that we like to call "warehouse machine learning tools." Examples of such tools are BigQuery ML and Continual. If you're already using BigQuery as your choice of data warehouse, then you can use BigQuery ML. Otherwise, check out Continual.

BigQuery ML

BigQuery ML is a component of BigQuery, Google's data warehouse offering. It lets you create and execute machine learning models in BigQuery using standard SQL queries. BigQuery ML empowers data analysts to use machine learning through existing SQL tools and skills.

Continual

Continual tags itself as "The missing AI layer for the modern data stack." It provides a simple workflow to build predictive models that can predict anything from customer LTV and customer churn to inventory forecasts and equipment failure, on top of your data warehouse. It also integrates nicely with dbt and can consume data sets directly from your dbt workflows.

MAKING THE MOVE TO WAREHOUSE-FIRST

Using a warehouse-first data analytics stack doesn't mean that you get rid of your basic analytics tools. They're still handy for basic analysis. However, the warehouse-first approach equips you with the armory to perform deeper analysis and answer questions that your analytics tool can't.

If your company is just starting out, a single analytics tool like Google Analytics or Amplitude will cater to the kind of questions you'll need answers to. But in the long term, you will outgrow them, and the sooner you set up your data warehouse, the better.