A Comprehensive study on Data Visualization tool - Grafana

1Sunil Kumar S H, 2Prof. Saravanan C
1Student, 2Assistant Professor
1Department of Master of Computer Applications,
2RV College of Engineering®, Bengaluru, India
sunilkumarsh.mca18@rvce.edu.in, sarvananc@rvce.edu.in

Abstract: As per research, currently, the world's data centers store about 1,327 exabytes of data and an average of 2.5 exabytes of data is produced every day. To analyze and understand this huge quantity of data is difficult. Data visualization is a pictorial representation or graphical representation of data that simplifies the understanding of data easier. To understand this data easily and represent this data on dashboards there are many visualization tools. Grafana is one tool which is very effective and efficient to visualize live streaming data such as weather data, stocks, websites incoming request and outgoing response and so on. Grafana is open-source visualization and analytics software. It allows you to query, visualize, alert on, and explore metrics no matter where they are stored. In general, it is a tool that converts the time series database data into beautiful graphs and visualizations which will be easier to analyze and understand. The paper also gives the comparison between the features of different type of data visualization tools like PowerBI, Tableau with Grafana and concludes with the result of when and why the Grafana data visualization tool needs to be used.

Index Terms – Data Visualization, Grafana, PowerBI, Tableau

I. Introduction

The data is measured in kilobytes, megabytes, gigabytes, or even terabytes till now. This is the amount of data an average person run across every day. In the coming years, however, these common units will begin to seem quaint – that’s because the entire digital universe has reached 44 zettabytes by 2020. By 2025, It is estimated that 463 exabytes of data will be generated each day which’s equivalent to 212,765,957 DVDs per day. Therefore, having a powerful and efficient data visualization tool to analyze, explore and represent the data visually is much needed. Grafana is open-source data analysis visualization and tool which allows visualizing, query, alerting and helps in exploring the metrics. Grafana is one of the efficient tools which efficiently visualize the live-streamed data directly into the dashboard with accurate results. Some of the key features of this tool are

• Visualize - Fast and flexible visualizations with multiple options that allow visualizing the data in any way the user want
• Dynamic Dashboard - Dynamic and reusable dashboards can be created using template variables
• Explore Metric -Ad-hoc queries and dynamic drill-down allows to explore the data. Splitting the view and comparing different time ranges and data sources parallelly
• Explore Logs - Can switch from metrics to logs with preserved label filters. Searches the logs very quickly and streams them live
• Alerting - Most of the important metrics will be alerted visually. The systems like Slack, PagerDuty, VictorOps, OpsGenie will be continuously evaluated and notifications will be sent by Grafana
• Mixed Data Source - The same graph can have different data sources. The data source can be selected based on the queries and it also works for custom data sources
• Annotations - Graphs with rich events can be annotated from different data sources. The events metadata and tags can be seen when hovered over events
II. Literature Survey

A picture is worth a thousand words, especially when dealing with exabytes of data. In current practice, many tools give the solution for data analysis through visualization such as Visme, Tableau, Power BI and so on. But all these tools are either for stored data or paid tools. And the tools which visualize live streaming data with efficient and accurate results are a handful.

There are many real-time application scenarios where the data should be visualized in real-time for efficient decision making. Weather data, Sensex data, Website health monitoring are some of the places where the live data needs to be analyzed to make the user understand the situations. The fusion of different data sources, showing changes and providing situational awareness are the three important problems, these problems lead to the main tasks of visual analysis of live streaming data, which are decision making, monitoring, trend and change detection, event tracking.

All the modern applications in today’s world represents one or the other visualization using graph or any other data representation. Hence it is important to display efficient and accurate visualization of raw data [1].

There are some techniques to create a meaningful data visualization. The size and composition play a very important role when selecting, representing and summarizing data. Analysis of data requires enough graphical representation, using 2D plots, hierarchical data and temporal data which helps in presenting data visually [2].

Even with the presence of enormous amount of data, it is not easy to understand the relationship between the data. In order to make a proper decision making without missing the important correlations the data visualization tools are used. With the use of data visualization tool any queries upon the data can be done easily and respective analysis can be made instantly so that, any person in an organization will be able to understand the data [3].

Auto charting is a data visualization technique that produces data visualization by just dragging and dropping the data you want to visualize. Auto charting will not create the exact visualization that the human has in mind [4].

Sankey diagrams make use of path analysis to show the movement of the transaction throughout the system. Sankey diagrams will display a series of linked nodes and the width of each lined nodes will represent the frequency or measure implied [5].

In the mobile devices data must be presented without losing the quality, in spite of whether it’s been used online or offline. Necessary design layouts must be considered to display the data in distinct screen shapes [6].

Google Analytics is a data visualization platform that organizes the product according to the way it is used. It shows the data associated with the respective products or service along with the links for product privacy and security setting [7].

Sisense is an analytical tool for small enterprises which provides market insights and analytical solutions for the business. Sisense is one among the few fully functional BI (Business Intelligence) software that let the non-technical user generate a dashboard, customize the dashboard and allows combining multiple datasets [8].

Tableau is a data visualization tool that allows a non-technical user to design the design time dashboards. Sharing of this dashboard does not require any programming knowledge [9].

Zoho Reports is an online award-winning business app provider. All the apps are hosted in cloud so that it can be used whenever needed by anyone [10].
III. Working of Grafana with Influx DB to monitor time series data

Influx DB is an open-source time-series database that is schema less and optional closed-source database developed by Influx Data. Influx DB is written in Go programming language for storing time series data. TICK Stack (Telegraf, Influx DB, Chronograf, Kapacitor) provides the services of Influx DB core and can run on any cloud and premises on a single node. Influx Cloud is the closed-source version of that is highly available, scalable, backup and restore. Deep insights and analysis, client library support and fast execution time are the important features of Influx DB.

Telegraf is an open-source server agent that helps in collecting all kinds of data from system, stacks and sensors. It’s a plugin designed for collecting metrics and as input and write them into array of outputs. As Telegraf is plugin-driven for collection of inputs and producing the output, it is extendable. Telegraf is written in Go programming language that is compiled and standalone binary that can be executed in any systems without the need of any external dependencies, no pip, no npm. or any other package management tool.

Grafana is one of the most popular time series data visualization tools which is highly recommended to use with Influx DB and Telegraf.

➢ Grafana Dashboard setup for Influx DB

For beginners, the first step is to download the influx DB and Grafana tools. The first basic setup is to have Grafana and Influx DB connected in the system. Influx DB will have API and the default port of Influx DB is 8086. Grafana will call the API of Influx DB whenever it needs to query the data and uses the default port has 3000.

Telegraf is the collection agent which collects all the metrics in Influx DB. There are over 200 plugins in Telegraf.

![Figure 2: Setup structure of Grafana with Influx DB and Telegraf](image)

Basic concepts to remember while setting up Grafana and Influx DB:

- Influx DB is a database that is mainly used for time series database. The Influx DB process can run on the same place where Grafana runs. Most of the Grafana monitoring happens in the browser and it is a very lightweight server-side application
- Setting up Grafana and Influx DB on the same instance is very easy, for very large installations of Influx DB, if the number of Grafana users are more or if the deployment profile is within the organization, then it is recommended to set up Grafana and Influx DB on a separate server which is perfectly acceptable
- Among Influx DB and Grafana, Influx DB will be more memory-intensive and CPU intensive, simply because a lot of Grafana’s work is done in the browser. It is recommended to use the latest version of the browser to get the best performance.

➢ Grafana and Influx DB setup Configurations

In Influx DB has a lot of default setup which can be used. Enabling log is one that logs all the queries when executed or when it is set to Influx DB API, which can be useful in solving Grafana debugging issues. The most important thing to remember when setting up Grafana with Influx DB is to set the coordinator section of config specifically, set the number of concurrent queries to the maximum. If different Grafana users are hitting the influx DB using different browsers and sending through a lot of database queries, it is recommended to set the maximum concurrent queries. The query time-out and set queries can be logged if it takes longer than a certain amount of time.

To limit the number of results that can be returned, the max-select point, max-select bucket and max-select series can be set, which prevents the expensive query from shutting down the Influx DB server or slow down for everyone else. The Grafana monitoring system can be made more usable by setting similar kind of configurations which includes router logging, HTTP ports and enabling the user to make the browser load faster.
Grafana Security Configurations

Every Grafana user has a default admin user and password. If the Grafana monitoring system is set on anywhere where anyone gets into the instance, it is recommended to set custom username and password to prevent unauthorized access from people. Here are some basic Grafana Security settings:

- By default, Grafana will allow the user to register and signup and also ask the non-admin user to create an organization. So, it is recommended to set the “enable anonymous access” option to “false” to prevent other people from setting up users in your organization if you don’t want them to.
- The Anonymous access is disabled by default, but it can be enabled if Grafana Dashboard is public which helps in promoting your work.
- If the password reset is done, which requires the user to receive an email, it should be set up in the SMTP (Simple Mail Transfer Protocol) section.
- Grafana has different log levels, so if the dashboard is being debugged, bump the log level to debug.
- The metrics in Grafana will expose itself. Telegraf will have Prometheus input which is built-in which can be directed towards and collect or receive internal Grafana metrics, store them in Influx DB and create the graphs again in Grafana.

Setting up Graphs for Grafana Metrics

Go to settings in Grafana home page and below configuration menu select and add a data source, select type of data source as Influx DB and give a name to the source. Select the Influx DB URL and enter the database name, user name and other details. Then once all the details are entered click on the “save and test” button and if the success message pops up, then Grafana Dashboard can be added, if the test result is “failed” then the entered details might be wrong and it needs to refilled again with correct data.

![Figure 3: Adding Influx DB data source to Grafana](image-url)

(a) Grafana home page, (b) Adding Influx DB data source, (c) Entering the details of data source, (d) Testing the connection of data source

In the Grafana Home page, select the dashboards menu and create a new dashboard, which redirects to the page where there is an option to add a row and panel. Grafana Dashboard is based on rows and panels, there can be multiple panels and multiple rows or a single row with one panel of specific visualization technique.
Figure 4: Creating Dashboard and adding panel to Grafana. (a) Creating new Dashboard from home page, (b) Adding panel and row for graph creation, (c) Filtering the data by querying in Influx DB

In edit mode, the Influx DB query is specified in the Grafana metrics path and also can specify the Influx DB policy that is used. You can set the different field and aggregation function to query, by default “value” is set as field and “mean” as aggregate function. In Grafana metrics, it is possible to select multiple fields or use the same field multiple times and use different aggregate functions to it. Grafana metrics allows you to add different transformation like moving, average, GROUPED BY is another property of Grafana metrics, by default Grafana group by will have a predefined interval value which is calculated depending on the panel and screen width. The user can specify Influx DB how it should handle null values. There are different ways of representing the graph (as lines, bars, points, charts etc.) and can have all this in a single dashboard too.

Figure 5: Setting up different metrics to create the graphs.

- **Optimize time series monitoring with Grafana**
  Grafana and Influx DB performs efficiently together, but you can hit some performance issues. If a huge set of time series data is collected or querying those data will put a lot of stress on Influx DB. This is common in Influx DB since Influx DB will do all the server-side works.

  Influx DB takes each Grafana graphs as queries. So, increase in the number of graphs will directly increase the load of the Influx DB server. If a Grafana Dashboard has 50 graphs, that’s 50 queries that are sent to Influx Db and 50 queries that need to get results collated by Influx Db and then sent back through Grafana monitoring. So, try limiting the number of graphs in the Dashboard so it increases the performance of both Grafana as well as Influx DB.

  Important points to remember to improve the performance of Grafana and Influx DB:
- Make your rows in Grafana Dashboard collapsible, so that the Grafana will not render all the graphs and it directly reduces the number of queries in the Influx DB. If there are many graphs that are not useful all at once, a great way to reduce the queries in Influx DB or speed up the Grafana Monitoring system with Influx DB is by using collapsible rows.
- There are many different graphs available and different options that can be added to the graphs. For example, if you concentrate on the number of queries executed, only the latest number of queries executed can be returned not graph over time. Templating and annotation can also be done instead.
- All the queries which are generated by Grafana graphs will have a GROUP BY clause, so a lot of recalculations is done on each loading of the graph.
- If the query is complex, the Influx DB will take a little more time to respond and the query may time out and an error message may appear in the Grafana Dashboard, wait for a little before hitting the refresh button.
- Both Grafana and Influx DB has different logging options. If an error message is displayed in Grafana graphs, go through the error message and figure out why the error is populated. The queries will not be cached in Influx DB, so whenever a query is sent it will be recalculated by Influx DB.
- The processing time of Grafana and Influx DB will depend on the amount of query that is passed. Typically, zooming out or in and selecting a large time range like 24 hours will take more time to respond.

![Figure 6: Grafana Dashboard](image.png)

**IV. Comparison of Features of different visualization tools**

It is often confusing for people for choosing the right tool for data visualization and analysis with visual insights and visually appealing. Here is the comparison of features of different visualization tools.

- **Usage**
  - **Grafana**: Used mainly for visualization of time series databases
  - **Power BI**: Used mainly for business intelligence or Analytics
  - **Tableau**: It is a business analytical tool that is used to create charts and interactive graphs in the form, of worksheets and dashboards

- **Visualization**
  - **Grafana**: In terms of data visualization, Grafana is rich in features, easy to use and very flexible. Grafana provides a wide range of custom options and visualization panels which gives life to data
  - **Power BI**: Provides the lesser amount of visualization when compared to Grafana data visualization
  - **Tableau**: Visualization is Tableau is done through dragging and dropping the measures and dimensions into different shelves with filters
Data Source

- **Grafana**: Grafana is compatible with time-series data sources like Influx DB, Graphite, Prometheus. It can easily integrate with different types of data source with minimum efforts.
- **Power BI**: Power BI is compatible with a data source which in the form database, file system or other online services. It primarily sticks to Microsoft products.
- **Tableau**: Tableau offers an extremely large number of data sources like MS Excel, PDFs, JSON or databases and servers like Tableau Server, MySQL Server, Microsoft SQL Server.

Filters

- **Grafana**: Grafana allows comparably lesser option to bring data directly into the dashboard.
- **Power BI**: Power BI has a wide variety of filters to dig deep into charts and panels.
- **Tableau**: Tableau provides a good filter option to add to the worksheets and dashboards.

Alerting

- **Grafana**: Currently in Grafana alerting is not supported for all the data source, but with a system like slack, pager duty, webhook events and mail notification alerting is integrated.
- **Power BI**: Alerting is allowed only through Power BI services with the option of either sending the email or adding a notification center.
- **Tableau**: Tableau has data-driven alerting system which automatically sends the notifications.

V. Results

Grafana is best suited for time-series data which can be visualized over charts or graphs. The numeric data that is generated from time-series data source are easy to query and manage using Grafana. The time-series data source can be integrated by using built-in support for the most prominent data sources which are available in the market. Grafana is the right choice when it comes to visualizing infrastructure, application, networks devices, sensors etc. It’s a great 24x7 monitoring solution for NOC and DevOps teams. It can also help to manage all the data from other application monitoring tools like AppDynamics, New Relic, Splunk, Dynatrace and gives an all-in-one web-based interface to display, alert and report over the data.

VI. Conclusion

This paper mainly concentrated on the Grafana visualization tool for streaming static as well as time-series data. The paper helps in understanding how to set up Grafana along with Influx DB and visualize data. It also shows how to create a new Influx DB data source and also creating a new Grafana Dashboard, extracting the data in the Influx DB. Knowing the importance of monitoring the time-series dashboard and Grafana security system which will be very helpful in creating efficient data visualization dashboards in Grafana. The paper also has the comparison between Grafana, Power BI and Tableau data visualization tool and draws a conclusion of when to use the Grafana Dashboards.

References

[2] Renato Toasa, Marisa Maximiano, Catarina Reis, David Guevara, “Data Visualization Technique for real time information – A Custom and Dynamic Dashboard for Analyzing Surveys Results”, Computer Science and Communication Research Centre (CIIC), Polytechnic Institute of Leiria, Portugal