# D3.JS

If you know Javascript, and you want to create images, the D3.JS library will make you happier than a 6-year-old with one of those real-ish lightsabers. D3.JS is one of the major Javascript data visualization libraries, and it’s used by some big-name players, including **The New York Times** and **Datameer**.

**Pros:**You can find excellents charts to display your data in a corporative dashbords

**Cons:**You need to have medium skills in javascript to integrate those compliment

**Overall:**I used this library in all web projects



D3.JS is a free JavaScript library that helps you create images using data. The tool enables you to connect arbitrary data to a Document Object Model (DOM), and then apply data-driven transformations to the document. With DOM programming API, programmers can access documents as objects.

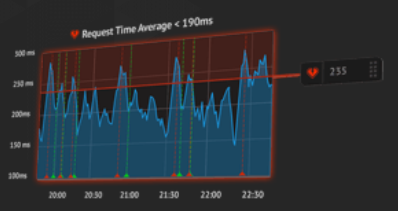
# GRAFANA

## Features

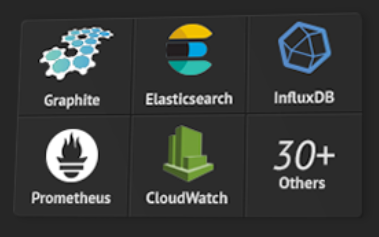
**Vissualize**: From heatmaps to histograms. Graphs to geomaps. Grafana has a plethora of visualization options

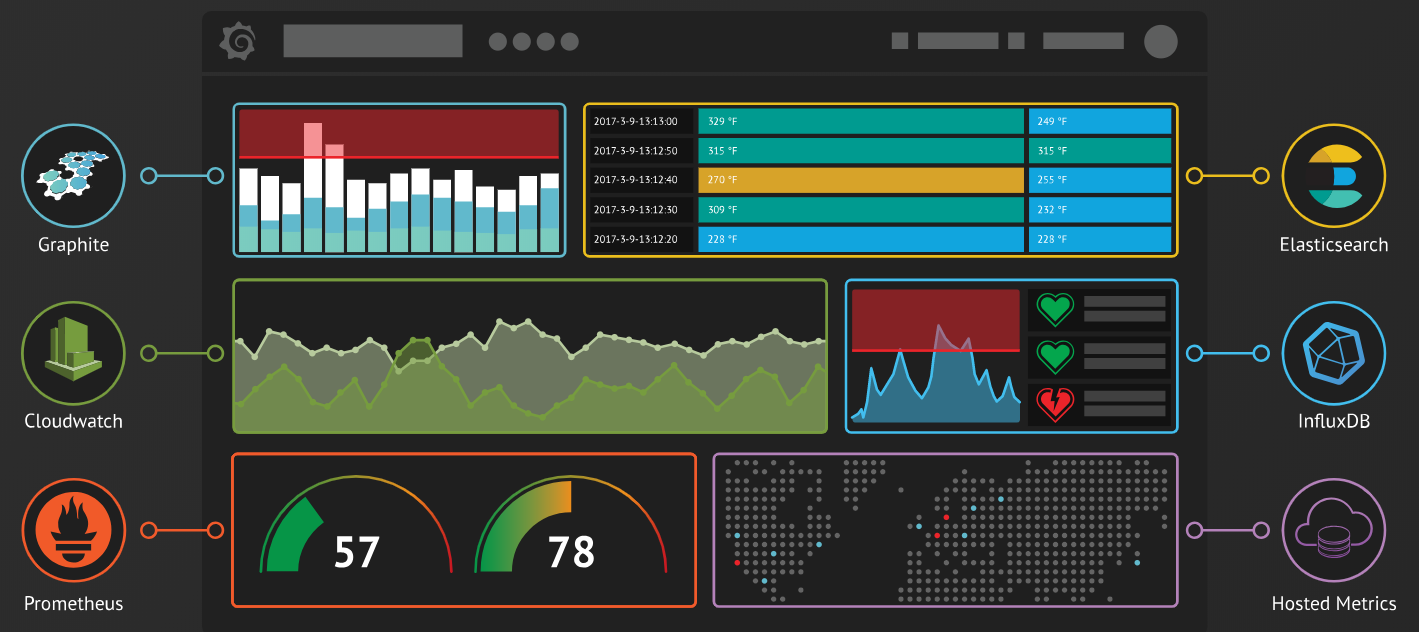


**Alert**: Define thresholds visually, and get notified via Slack, PagerDuty, and more



**Unify**: Grafana supports dozens of databases, natively. Mix them together in the same Dashboard.





No matter where your data is, or what kind of database it lives in, you can bring it together with Grafana.

**Open**: Open source, you can install on any platform



**Extend**: Discover hundreds of dashboards and plugins in the official library.



**Collaborate**: share data and dashboards across teams. Share link to dashboard or fullscreen panel. Automatically includes current time range and variables. Create a public or internal snapshots.



## Plugins:

**PANEL**

|  |  |  |  |
| --- | --- | --- | --- |
| **World Map Panel:** displays time series data or geohash data from Elasticsearch overlaid on a world map. | [PictureIt Logo](https://grafana.com/plugins/bessler-pictureit-panel)  **Picturelt:** Add Measurements to a Picture in Grafana | **Annunciator:** Enhanced versin of built-in SingleStat panel, with specialized display of thresholds and value-sensative... | **Diagram Panel** |
| [**Clock** **Panel**](https://grafana.com/plugins/grafana-clock-panel) | **Table Panel for Grafana** | **Boom table panel for Graphite, InfluxDB, Prometheus** | **[Discrete](https://grafana.com/plugins/natel-discrete-panel)** [Events grafana](https://grafana.com/plugins/natel-discrete-panel) |
| **[Pie Chart Logo](https://grafana.com/plugins/grafana-piechart-panel)**  **Pie chart panel** | **Text** | **Breadcrumb Panel for Grafana** | **[epict Panel Logo](https://grafana.com/plugins/larona-epict-panel)**  **epict Panel:** Enter the URL of the image you want, and add some metrics on it. |
| **Alert List:** Shows list of alerts and their current status | [A Cesium based](https://grafana.com/plugins/satellogic-3d-globe-panel) **[3D Globe panel](https://grafana.com/plugins/satellogic-3d-globe-panel)** [plugin.](https://grafana.com/plugins/satellogic-3d-globe-panel) | **[Bubble Chart Logo](https://grafana.com/plugins/digrich-bubblechart-panel)**  **Bubble Chart: Bubblechart panel** | **[GeoLoop Logo](https://grafana.com/plugins/citilogics-geoloop-panel)**  **GepLoop Looping animated map** |
| **Dashboard List:** List of dynamic links to other dashboards | **AJAX panel for grafana** | **[Cal-HeatMap Logo](https://grafana.com/plugins/neocat-cal-heatmap-panel)**  **Cal-Heat Map: Cal-HeatMap panel for Grafana** | **Heatmap panel** |
| **Graph Panel** | **[Alarm Box Logo](https://grafana.com/plugins/btplc-alarm-box-panel)**  **Alarm Box: Box panel counting values in a series** | **Carpet plot panel plugin** | **WindRose:** Make windrose plots |
| **Heatmap Panel** | **Annotation List:** [List of builtin Annotations](https://grafana.com/plugins/ryantxu-annolist-panel) | **D3 Gauge:** D3-based Gauge panel for Grafana | **HeatmapEpoch** |
| **Plugin List** | **[Annotation Panel Logo](https://grafana.com/plugins/novalabs-annotations-panel)**  **Annotation Panel** | **[Datatable Panel Logo](https://grafana.com/plugins/briangann-datatable-panel)**  **Datatable Panel** | **Histogram Panel** |
| **InfluxDB admin** | **Organisations Panel** | **[Parity Report Logo](https://grafana.com/plugins/zuburqan-parity-report-panel)**  **Parity report plugin to compare metrics** | **[Peak Report Logo](https://grafana.com/plugins/btplc-peak-report-panel)**  **Peak Report: Table panel showing peaks across series** |
| **Singlestat Panel** | **Plotly:** Scatter plots and more | **Polystat panel** | **[Radar Graph Logo](https://grafana.com/plugins/snuids-radar-panel)**  **Radar Graph** |
| **[[Trend Box Logo](https://grafana.com/plugins/btplc-trend-box-panel)Trend Box](https://grafana.com/plugins/btplc-trend-box-panel)** [Box panel showing trends over series](https://grafana.com/plugins/btplc-trend-box-panel) | **Status By Group Panel** | **[Status Dot Logo](https://grafana.com/plugins/btplc-status-dot-panel)**  **Status Dot:** Dot panel showing status across series | **Status Panel** |
| **Statusmap panel** | **[[SVG Logo](https://grafana.com/plugins/marcuscalidus-svg-panel)](https://grafana.com/plugins/marcuscalidus-svg-panel)**  **[SVG panel](https://grafana.com/plugins/marcuscalidus-svg-panel)** | **[Traffic Lights Logo](https://grafana.com/plugins/snuids-trafficlights-panel)**  **Traffic Lights:** | **[TrafficLight Logo](https://grafana.com/plugins/smartmakers-trafficlight-panel)**  **TrafficLight:** Add colour indicator for measurements to a picture in Grafana |

**DATA SOURCE**

|  |  |  |  |
| --- | --- | --- | --- |
| **[CloudWatch Logo](https://grafana.com/plugins/cloudwatch)**  **Cloudwatch** |  | **PostgreSQL** | **DeviceHive** |
| **ElasticSearch** | **[Stackdriver Logo](https://grafana.com/plugins/stackdriver)**  **Google Stackdriver** | **Cloudera Manager** | **[Oracle Database Logo](https://grafana.com/plugins/grafana-oracle-datasource)**  **Oracle Database** |
| **[Graphite Logo](https://grafana.com/plugins/graphite)**  **Graphite** | **Amazon Timestream** | **Consul** | **[Druid Logo](https://grafana.com/plugins/abhisant-druid-datasource)**  **Druid** |
| **InfluxDB** | **[Ambari Metrics Logo](https://grafana.com/plugins/praj-ams-datasource)**  **Ambari Metrics** | **[Crate Logo](https://grafana.com/plugins/crate-datasource)**  **Crate SQL Database** | **[Dynatrace Logo](https://grafana.com/plugins/grafana-dynatrace-datasource)**  **Dynatrace** |
| **[Akumuli Logo](https://grafana.com/plugins/akumuli-datasource)**  **Akumuli** | **Appdynamics** | **[DalmatinerDB Logo](https://grafana.com/plugins/dalmatinerdb-datasource)**  **DalmaterinDB** | **Finance** |
| **[OpenTSDB Logo](https://grafana.com/plugins/opentsdb)**  **OpenTSDB** | **[Azure Data Explorer Datasource Logo](https://grafana.com/plugins/grafana-azure-data-explorer-datasource)**  **Azure Data Explorer** | **DarkSky** | **Flux(InfluxDB)** |
| **Gnocchi** | **[Azure Monitor Logo](https://grafana.com/plugins/grafana-azure-monitor-datasource)**  **Azure Monitor/Application Insights** | **[Datadog Logo](https://grafana.com/plugins/grafana-datadog-datasource)**  **Datadog** | **GoogleCalendar** |
| **Prometheus** | **[Blueflood Logo](https://grafana.com/plugins/rackerlabs-blueflood-datasource)**  **blueflood** | **Heroic** | **Hawkular** |
| **[IBM APM Logo](https://grafana.com/plugins/ibm-apm-datasource)**  **IBM APM** | **[InfluxDB 0.8.x Logo](https://grafana.com/plugins/grafana-influxdb-08-datasource)**  **InfluxDB 0.8.x** | **[Instana Logo](https://grafana.com/plugins/instana-datasource)**  **Instana** | **JSON** |
| **[KairosDB Logo](https://grafana.com/plugins/grafana-kairosdb-datasource)**  **KairosDB** | **[LinkSmart HDS Logo](https://grafana.com/plugins/linksmart-hds-datasource)**  **LinkSmart HDS** | **LinkSmart Sensor Things** | **[MetaQueries Logo](https://grafana.com/plugins/goshposh-metaqueries-datasource)**  **MetaQueries** |
| **[Monasca Logo](https://grafana.com/plugins/monasca-datasource)**  **Monasca** | **Monitoring Art** | **[NetXMS Logo](https://grafana.com/plugins/radensolutions-netxms-datasource)**  **Net XMS** | **New Relic** |
| **Ntopng** | **[Open-Falcon Logo](https://grafana.com/plugins/fastweb-openfalcon-datasource)**  **Open-Falcon** | **[openHistorian Logo](https://grafana.com/plugins/gridprotectionalliance-openhistorian-datasource)**  **openHistorian** | **ClickHouse** |
| **[OSIsoft-PI Logo](https://grafana.com/plugins/gridprotectionalliance-osisoftpi-datasource)**  **OSIsoft-PI** | **PNP** | **Prometheus Alert Manager** | **[PRTG Logo](https://grafana.com/plugins/jasonlashua-prtg-datasource)**  **PRTG** |
| **[Sidewinder Logo](https://grafana.com/plugins/sidewinder-datasource)**  **Sidewinder** | **[Skydive Logo](https://grafana.com/plugins/skydive-datasource)**  **Skydive** | **[Splunk Logo](https://grafana.com/plugins/grafana-splunk-datasource)**  **Splunk** | **[Sun and Moon Logo](https://grafana.com/plugins/fetzerch-sunandmoon-datasource)**  **Sun and Moon** |
| **[Warp10 Logo](https://grafana.com/plugins/ovh-warp10-datasource)**  **Warp10** | **USGS Water Services** |  |  |

# OMNİSCOPE

Pros:   
  
- All-in-one software tool for ETL and publishing reports.   
  
- Lots of useful data feeds for adservers reports, ESRI ArcGIS data, etc

- the data manager is a dream to work with, the ease of visualisation in the desktop (designer app) is at the top of its game, the mobile version is very flexible and quick and finally the after sales support is top notch.

Cons:   
  
- Not yet cloud deployed   
- To see output you need to install the free viewer at least

-limited security model for web

**FEATURES**

**Everything**: Data blending, analytics, visualisation, reporting.

**Scalable**: Prepare big data, fast.

**Everywhere**: iOS, Android, Windows, Mac, Linux.

**Open**: Custom R and Python analytics, and JS visualisations.

**Simple**: Zero configuration install. Touch-driven interface.

**Building blocks for data blending**

Omniscope's pluggable blocks let you explore and build your data pipeline.

Extract and process big data. Transform, combine and clean. Use in Omniscope's visualisations, or export to another system.

**Data sources include:**

* Hadoop, Impala
* Excel, CSV, JSON, XML, text
* SQL databases
* Actian Vector
* Google Analytics
* Amazon Redshift
* Reddit
* Twitter
* Instagram
* DoubleClick

**Intelligent analytics**

Omniscope helps you do advanced analytics the right way.

* Clustering
* Prediction
* Regression
* Forecasting
* Seasonality
* Topic modelling
* Basket analysis
* Network analysis

Other advanced analytics include:

* Custom R
* Covariance/correlation
* Custom Python
* Dimensionality reduction
* One Way ANOVA
* Oversampling
* Variable Importance

# BANANA

Banana is a data visualization tool that uses Solr for data analysis and display.

Data display in Banana is based on dashboards, which contain rows of panels that implement the analysis required.

Banana is open source, and based on a port of Kibana. The source code is available in Github at <https://github.com/Lucidworks/banana/>.

The Banana dashboard is the central feature of Banana, and is the place where the various visualizations are stored.

A dashboard contains one or more controls for search query inputs and one or more quantitive displays over the results for that query. These controls and displays are called panels.

Dashboards run as a client-side application in a web browser. Solr facets provide the quantifications required for visualizations, which can be charts, graphs, tables, and maps (for geospatial data). Dashboards also have tabular displays for drilling down to the individual documents in a results set.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Features | Grafana | Omniscope | D3.js | Banana |
| Open Source | + | + | + | + |
| Data Sources |  | * Hadoop, Impala * Excel, CSV, JSON, XML, text * SQL databases | The CData API Server enables you to generate REST APIs for 80+ data sources, including both on-premises and cloud-based databases. |  |
| Extendable | Different plug-ins |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Built on | most commonly used together with Graphite, InfluxDB, and also Elasticsearch and Logz.io. | ODBC(one database, multiple tables) & JDBC (multiple databases, multiple tables) - two standard protocols for connecting to relational databases, these options are sufficient for connecting to SQL-based transactional tables or reporting views. | The CData API Server | Solr |
| Easy to use |  |  | You need to have medium skills in javascript to integrate those compliment.  More build on web app with html, css. | Visualizations. It is easy to create visualizations after the data is loaded. The automatic update of the graphs from a dashboard |

## Kibana

[Kibana](https://logz.io/blog/kibana-tutorial/) is the ‘K’ in the [ELK Stack](https://logz.io/learn/complete-guide-elk-stack/), the world’s most popular open source log analysis platform, and provides users with a tool for exploring, visualizing, and building dashboards on top of the log data stored in Elasticsearch clusters.

Kibana’s core feature is data querying and analysis. Using various methods, users can search the data indexed in Elasticsearch for specific events or strings within their data for root cause analysis and diagnostics. Based on these queries, users can use Kibana’s visualization features which allow users to visualize data in a variety of different ways, using charts, tables, geographical maps and other types of visualizations.

## Grafana

[Grafana](http://grafana.org/) is an open source visualization tool that can be used on top of a variety of different data stores but is most commonly used together with Graphite, InfluxDB, and also Elasticsearch and Logz.io.

Essentially, it’s a feature-rich replacement for [Graphite-web](https://github.com/graphite-project/graphite-web), which helps users to easily create and edit dashboards. It contains a unique Graphite target parser that enables easy metric and function editing. Users can create comprehensive charts with smart axis formats (such as lines and points) as a result of Grafana’s fast, client-side rendering — even over long ranges of time — that uses [Flot](http://www.flotcharts.org/) as a default option.

## Logs vs. metrics

The key difference between the two visualization tools stems from their purpose. Grafana is designed for analyzing and visualizing metrics such as system CPU, memory, disk and I/O utilization. Grafana does not allow full-text data querying. Kibana, on the other hand, runs on top of Elasticsearch and is used primarily for analyzing log messages.

If you are building a monitoring system, both can do the job pretty well, though there are still some differences that will be outlined below. If it’s logs you’re after, for any of the use cases that logs support — troubleshooting, forensics, development, security, Kibana is your only option.

## Data sources and integrations

Grafana was designed to work as a UI for analyzing metrics. As such, it can work with multiple time-series data stores, including built-in integrations with Graphite, Prometheus, InfluxDB, MySQL, PostgreSQL, and Elasticsearch, and additional data sources using plugins. For each data source, Grafana has a specific query editor that is customized for the features and capabilities that are included in that data source. Kibana on the other hand, is designed to work only with Elasticsearch and thus does not support any other type of data source.

## Access control and authentication

bana dashboards are open and accessible to the public. In comparison, Grafana ships with built-in user control and authentication mechanisms that allow you to restrict and control access to your dashboards, including using an external SQL or LDAP server. In addition, Grafana’s API can be used for tasks such as saving a specific dashboard, creating users, and updating data sources. You can also create specific API keys and assign them to specific roles.

**Querying**

With Grafana, users use what is called a Query Editor for querying. Each data source has a different Query Editor tailored for the specific data source, meaning that the syntax used varies according to the data source. Graphite querying will be different than Prometheus querying, for example.

## Dashboards and visualizations

Both Kibana and Grafana boast powerful visualization capabilities. Kibana offers a rich variety of visualization types, allowing you to create pie charts, line charts, data tables, single metric visualizations, geo maps, time series and markdown visualizations, and combine all these into dashboards. Dashboards in Kibana are extremely dynamic and versatile — data can be filtered on the fly, and dashboards can easily be edited and opened in full-page format. Kibana ships with default dashboards for various data sets for easier setup time.

Grafana dashboards are what made Grafana such a popular visualization tool. They are infamous for being completely versatile. Visualizations in Grafana are called panels, and users can create a dashboard containing panels for different data sources. Grafana supports graph, singlestat, table, heatmap and freetext panel types. Grafana users can make use of a large ecosystem of ready-made dashboards for different data types and sources.

Functionality wise — both Grafana and Kibana offer many customization options that allow users to slice and dice data in any way they want. Users can play around with panel colors, labels, X and Y axis, the size of panels, and plenty more. All in all though, Grafana has a wider array of customization options and also makes changing the different setting easier with panel editors and collapsible rows.

Both Kibana and Grafana are powerful visualization tools. However, at their core, they are both used for different data types and use cases. Grafana together with a time-series database such as Graphite or InfluxDB is a combination used for metrics analysis,  whereas Kibana is part of the popular ELK Stack, used for exploring log data.