Sensu

~By sumo logic

## **Introduction:**

Sensu is a monitoring tool for ephemeral infrastructure and distributed applications that is open source. It is an agent-based monitoring system with auto-discovery built-in, making it ideal for cloud environments. Sensu monitors service health and collects telemetry data using service checks. It also has a number of well-defined APIs for configuration, external data input, and data access. Sensu, also known as "the monitoring router," is extremely extensible.

## **About Sensu:**

Sensu's monitoring event pipeline enables businesses to automate monitoring workflows and gain comprehensive visibility into their multi-cloud infrastructure, from Kubernetes to bare metal. Sensu is used by companies such as Sony, Box.com, and Activision to help them deliver value faster and at scale.

## **What is Sensu Go?**

Sensu has been completely rewritten in Go, adding new features and lowering operational overhead. It removes a number of friction points for both new and seasoned Sensu users.

External services like Redis or RabbitMQ were necessary for the original Sensu. Sensu Go's ability to rely on an inbuilt etcd datastore for persistence makes the software simpler to use. If you already have them installed, you can also use external etcd services.

Sensu Go replaces Ruby expressions with JavaScript filter expressions by incorporating a JavaScript interpreter.

Sensu Go events are always handled, unlike the original Sensu, unless they are specifically filtered.

## **Project Summary**

| Website | <https://sensu.io/> |
| --- | --- |
| Organization/Foundation name | Sensu.inc, A venture-funded company backed by Battery Ventures and Foundry Group. Sensu Inc was acquired in June 2021 by [Sumo Logic Inc](https://www.sumologic.com/). |
| License | MIT license |
| open/proprietory | Based on open source core that is freely available under a permissive MIT license and publicly available on GitHub. |
| Source path/ project repo | <https://github.com/sensu/sensu-go> |
| Organization GitHub repo | <https://github.com/sensu> |
| Brief description | Sensu Go is a commercially available monitoring and observability solution for enterprises, empowering DevOps and SRE teams to automate their monitoring workflows and gain deep visibility into their infrastructure – from Kubernetes to bare metal. |
| Documentation link | <https://docs.sensu.io/sensu-go/latest/> |

* **Project details:**

### **# Key features:**

* Filtered,context-rich alerts that improve incident response.
* Extend functionality and integrate with existing workflows with the Sensu Catalog.
* Automate with agent registration-deregistration and check subscriptions
* Built-in support for industry-standard tools
* Intuitive API with command line and web interfaces
* Commercial software based on the open core.

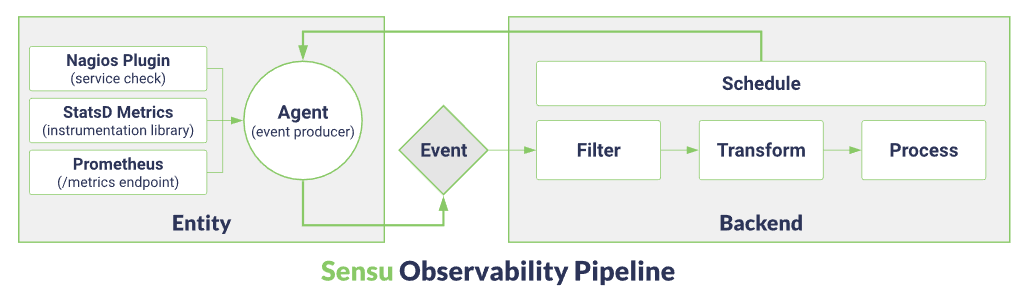
To know more about the features visit <https://sensu.io/features>

### **# Architecture:**

Sensu Observability pipeline

* The Sensu agent is a lightweight process that runs on the infrastructure components you want to observe.
* The Sensu backend schedules check for agents to run on your infrastructure.
* The agent runs these checks on your infrastructure to gather observation data about your networking, compute resources, applications, and more.
* Events contain the observation data that your checks gather, which might include entity status, metrics, or both, depending on your needs and configuration.
* For example, Sensu can send entity status data in an email, Slack, or PagerDuty alert and transport metrics to storage in your Graphite, InfluxDB, or Prometheus databases.

*Architecture diagram:*



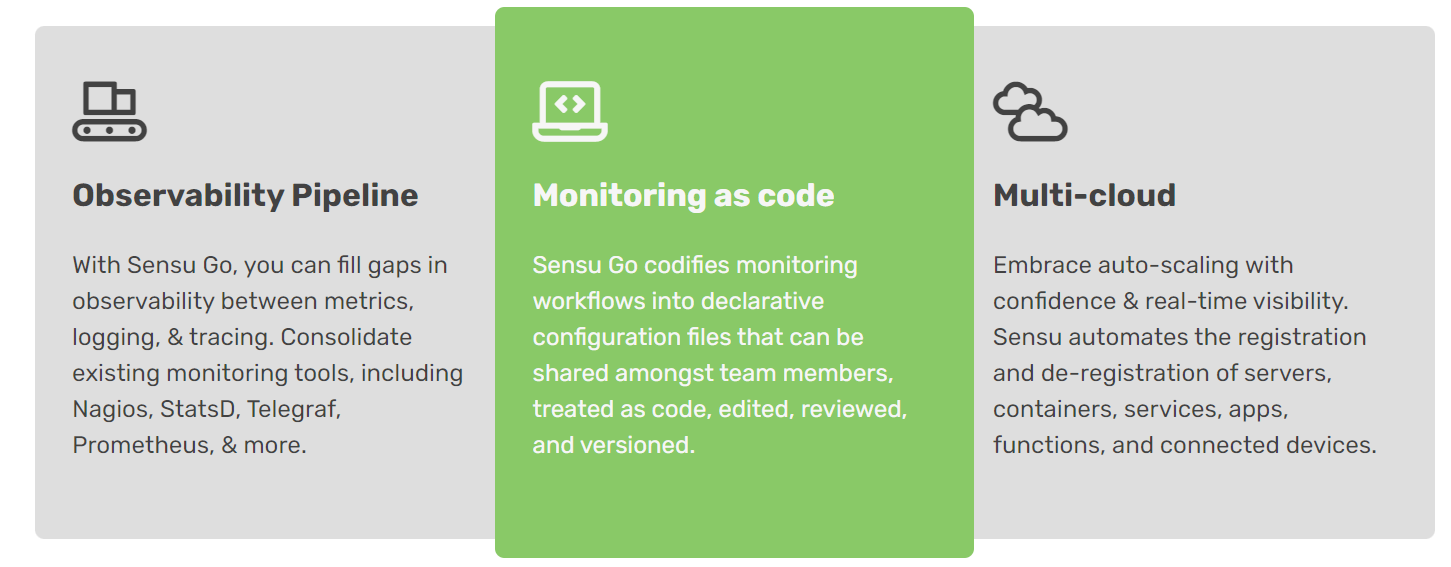
### **# Current Usage:**

Trusted for production workloads by industry leaders like



## **Why choose Sensu?**

It is necessary to switch from a host-based to a functional role-based monitoring strategy as a result of the transition from static to dynamic infrastructure. The control plane transitions from point-and-click interfaces to infrastructure as code workflows and self-service developer APIs, while connectivity switches from remote polling to publish-subscribe. To know the use cases visit <https://sensu.io/use-cases>



### **# Technical details/Supported platforms and distribution**

<https://docs.sensu.io/sensu-go/latest/platforms/>

### **# project integrations:**

<https://sensu.io/features/monitoring-integrations>

Reference/Acknowledgment

Website : <https://sensu.io/>

Documentation :<https://docs.sensu.io/sensu-go/latest/>

Github :<https://github.com/sensu/sensu-go>

Community Forum : <https://discourse.sensu.io/>

Resources :<https://sensu.io/resources>

Slack:<https://sensucommunity.slack.com/join/shared_invite/zt-1njwaoous-lFHx7hPZtzwc2RNupwC6Mg#/shared-invite/email>