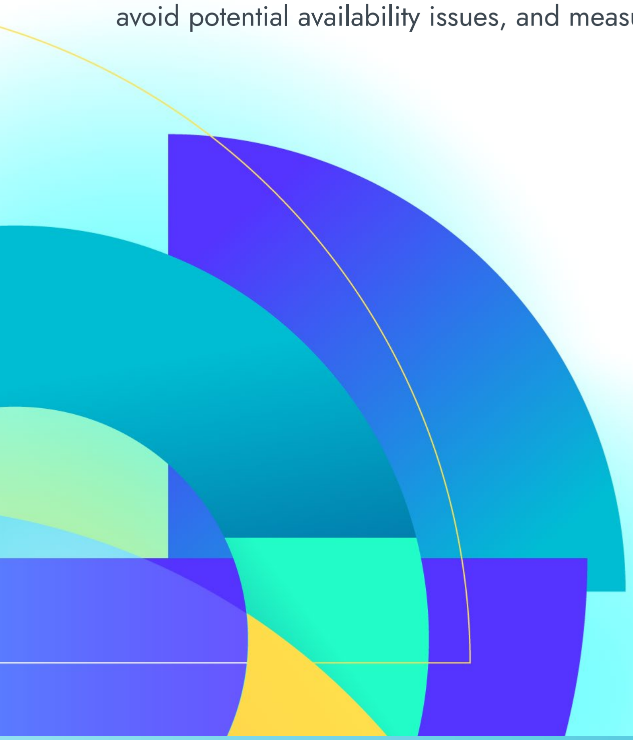
sysdig

Best Practices for Reducing the Cost of Custom Metrics

When it comes to Kubernetes and cloud monitoring, users must be sure that not only the cloud infrastructure is under control, but that their own applications and other third-party applications exposing metrics themselves are properly observed. Here is when custom metrics come into play.

Talking about Kubernetes and cloud monitoring, custom metrics are a key factor for a huge number of companies. Custom metrics refer to data points that are specific to a particular business or application, and are not typically captured by standard monitoring tools. These metrics provide valuable insight into the performance and usage of a specific application or service, and can help identify areas for improvement or optimization. They must rely on solid observability systems to watch performance, avoid potential availability issues, and measure their own business KPIs. That's why custom metrics are normalized and widely adopted across almost every organization. But, what about the associated costs of maintaining and storing all these custom time series metrics? As you'd probably guess, this can cause huge overspending.

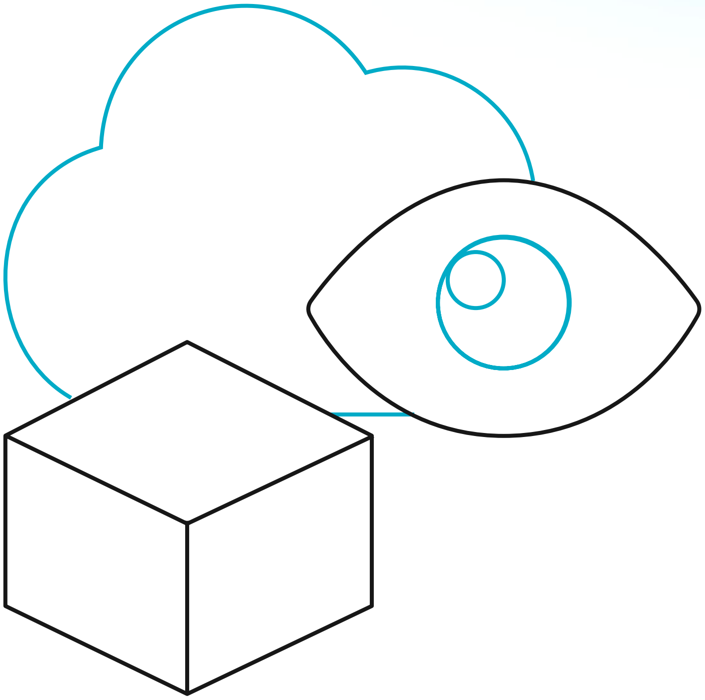


When talking about metrics costs, metrics cardinality is important. But, what does cardinality mean? The definition of cardinality is "The number of elements in a given mathematical set." Metrics can have multiple labels, at the same time labels can have many different values. The more values for a label, and the more labels in a metric, the more cardinality. In a metrics context, we can say cardinality is the number of combinations of labels and its values for a metric. If you add a new key/value pair to your metric, you will be exponentially increasing cardinality. The more cardinality, the more unique metrics. That means more storage and more infrastructure is needed to support and process data ingestion, storage, and metrics exploration. In summary, this means increased costs.

In this best practices guide, you'll find some tips that will help you with reducing the cost of custom metrics monitoring.

1

Choose your cloud-native observability platform wisely



The observability platform choice is undoubtedly a tough decision. There are several offerings on the market, from a DIY Prometheus to a bunch of open source tools and add-ons, like exporters, cadvisor, KSM, AlertManager, Grafana, etc., to a wide variety of SaaS observability platforms. It is up to you to decide which option fits best your needs.

While companies grow and keep progressing on cloud-native adoption, Kubernetes and cloud costs grow as well. Cloud-native observability platforms are not an exception. The more applications and services, the more metrics, and more associated costs to store, maintain, and explore data.

The following quote from a cloud-native observability user, is a clear example of choosing the right observability platform.

> **“As we scaled, Datadog was becoming increasingly painful to work with. They started putting limitations on what we could emit, causing them to block reporting and metrics. As a result, metrics would disappear. We didn’t know it until we needed the metric, and by then, it was too late.”**

Let's talk about some key factors to consider when reducing costs of cloud-native observability platforms for your business.

- Explore the options available on the market. There are many solutions available at different price points. When comparing custom metrics prices between providers, you'll find huge differences, up to 10x more expensive for ingesting the same amount of metrics.

- Analyze custom metrics costs carefully, not only the short-term costs but the long run as well. Some providers charge additional fees when surpassing certain thresholds. Your bill can grow exponentially over time.

- The number of containers per host and events per host can be charged in some observability services. Identify how much money you will spend here based on your numbers. There are solutions in the market offering no extra charges per number of containers or events.

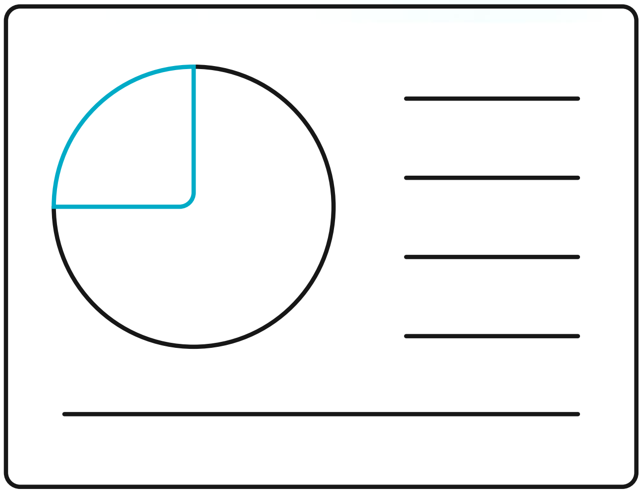
- Tagging metrics is a super powerful feature, but depending on how this is handled under the hood, it may mean a cardinality explosion. When companies start using custom metrics, they start collecting data, labels are filled with different values, then cardinality grows. In short, a metric + tag is a new metric, so the number of metrics can grow dramatically. For applications and services where metrics grow fast and the exponential magnitude is high, then there is what is commonly called cardinality explosion.

- The number of custom metrics supported vary between offerings available on the market. Plan accordingly and be sure your requirements are met. You don't want to reach certain limits and miss part of your data. The more metrics the more costs.

- Not all the solutions scale well. In terms of performance, you may find difficulties to manage and analyze your data when a certain number of metrics is reached. How many custom metrics can you save? What about the retention period? Is there any limitation when storing and accessing my data? What is the scrape interval period? These are some of the questions you can ask. Sysdig Monitor is the way to use Prometheus at scale with custom metrics.

2

Improve custom metrics visibility



Visibility is the foundation of any observability platform. When it comes to custom metrics, it is key to be able to select and filter which metrics you want to ingest and store. As it was already mentioned, more metrics means more storage is needed, and more costs for monitoring and analyzing your data. It is quite common to avoid spending time on thorough examination of what is really needed and what can be discarded. But the default scenario is not always the most efficient in terms of cost.

Let's go through some points that will help you save costs when talking about custom metrics visibility.

- Your cloud-native observability platform has to be able to provide visibility on your custom metrics usage. Have you reached the limits? How much room do you have to keep growing in terms of metrics volume over time?

- Are some metrics being dropped? Ensure you can easily check if that happens at any point.

- Ensure you have control on the custom metrics volume coming from remote write endpoints. Depending on your application infrastructure, you may have several sources pushing metrics without control. Do you want to know more about remote write for custom metrics?

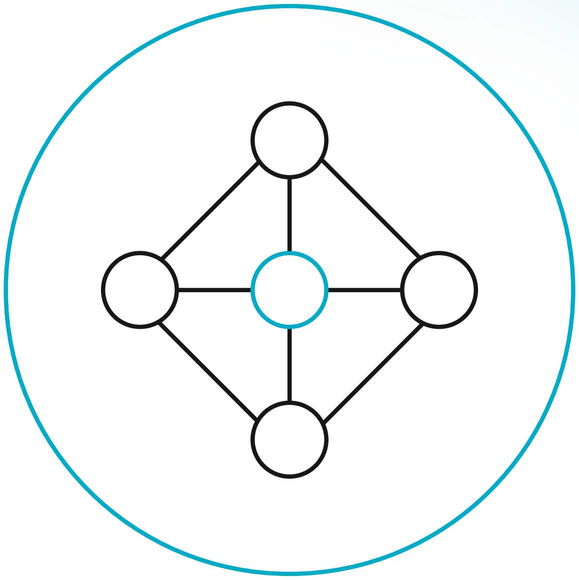
- Monitor the custom metric volume by host. It will give you insights on the host volume balance, avoiding performance bottlenecks.

- Not only Prometheus custom metrics can be ingested. What about JMX, StatsD, or applications themselves exposing metrics? Be sure you have enough visibility to differentiate among different types of metrics.

- The number of containers being monitored can be another factor for making your bill grow at the end of the month. Monitor the number of containers you host to avoid unnecessarily increasing your costs.

3

Collect and store only the metrics you need



As the title indicates, you should only collect and store the metrics that you need. It sounds obvious, but most businesses are managing more metrics than they really need.

- When monitoring your Kubernetes and cloud infrastructure, you may end up ingesting tons of metrics that you may not need. The same happens with custom metrics. This can cause a negative impact not only on your bill, but on the performance of your observability system. Collect information and evaluate which metrics you are pulling from your environments, and start triaging.

- Select only and filter by the application metrics that are truly essential for your business. The more application metrics the more costs. This will help you to reduce wasted spending, and make it easier to manage and analyze the data.

- Are you using your own backend? If so, be sure you are using the appropriate storage. It has to perform well to avoid issues while managing and analyzing your data. On the other hand, it needs to fit your budget, otherwise you'll run into problems as soon as your custom metrics volume starts growing. Cloud long-term storage like the AWS S3 can help you reduce your storage costs.

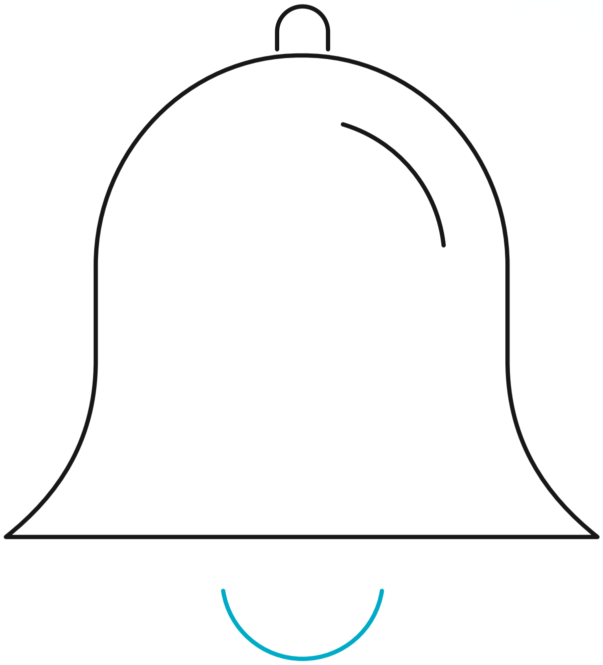
- Cloud-native observability SaaS providers usually provide their own long term storage service as a part of their SaaS. When choosing an observability platform, checking long-term storage prices and conditions for custom metrics is a must. That way, you'll avoid a lot of pain in the form of huge bills.

4

Custom metrics alerting

Alerts can help warn you about a service not being responsive, when throughput is too high or low, or even when a certain business condition is met. These are some of the most common patterns when setting alerts for our environments.

But, what about alerts for your custom metrics? How can you benefit from alerting systems when managing high volumes of custom metrics?



- Set alerts on your custom metrics usage. Be alerted when custom metrics volume reaches a certain threshold. As discussed in previous sections, some cloud-native observability platforms may overcharge you when exceeding a certain limit. Take action and avoid huge overspending.

- Alerting on the number of custom metrics you are processing is also beneficial to avoid missing custom metrics in your platform. Set alerts on limits and avoid unnoticed metrics drops.

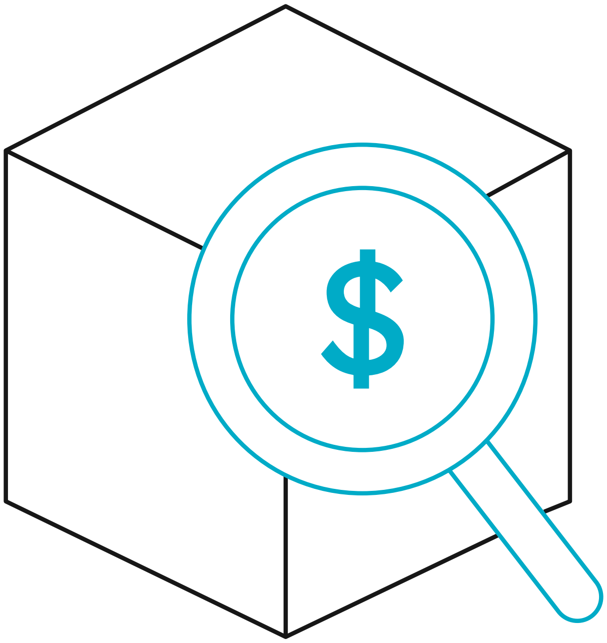
- Also consider setting alerts on the number of metrics per node to avoid surpassing the custom metric limits per node.

- In order to know how your environment and applications are evolving, in terms of custom metrics volume and costs, monitor and set alerts by metrics type (Prometheus, applications, StatsD, JMX). You can even dig deeper and set alerts by Prometheus jobs, or some other tag of your choice.

5

Custom metric costs management

Controlling Kubernetes and cloud costs are a big deal for companies that are on their cloud-native journey, but can certainly be challenging. When it comes to observability platforms, custom metrics can cause significant overspending. It can be costly to implement and maintain because they require additional resources, such as servers and storage, and may also require specialized software and expertise.



- Some cloud-native observability platforms allow you to monitor and manage the costs of your custom metrics, Kubernetes, and cloud infrastructures. Take the burden off and let the observability platform do this hard job for you. Sysdig Cost Advisor can help you with monitoring and managing your costs.
- Identify the biggest overspending areas in your Kubernetes and cloud environments by cluster, namespace, workload, or Pod.
- Keep your costs under control and take action before costs are a big pain for your business.
- Elaborate your own cost reports, feeding your chargeback tool with information from your infrastructure. It will allow you to identify, distribute, and own costs by different business units or departments within your organization.

Sysdig Monitor radically simplifies cloud and Kubernetes monitoring and helps lower costs with deep visibility into cloud-native workloads. You get immediate, granular details and troubleshooting tools for rapidly changing container environments. Our cost savings estimates based on utilization metrics help you prioritize rightsizing efforts. Offload the maintenance of Prometheus servers and get a fully compatible managed service with long-term storage, automatic service detection, out-of-the-box dashboards/alerts, and curated exporters for integrations. If you need custom metrics, Sysdig Monitor can provide more data from your application environments at a lower cost.

www.sysdig.com

REQUEST PERSONALIZED DEMO

Guide-016 Rev. A 2/23