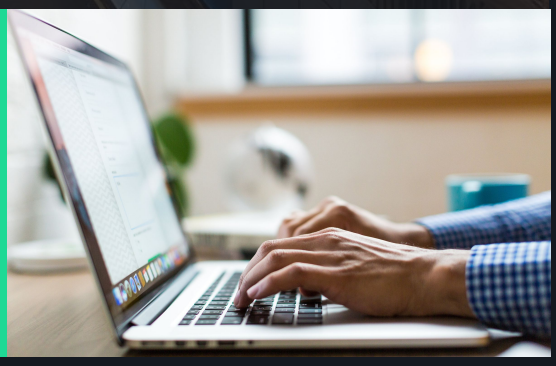
chronosphere

WHITEPAPER

Prometheus-Native Monitoring SaaS Solutions Buyers Guide

chronosphere.io

Ready to stop managing your own Prometheus? Here's your buyers guide



Executive summary

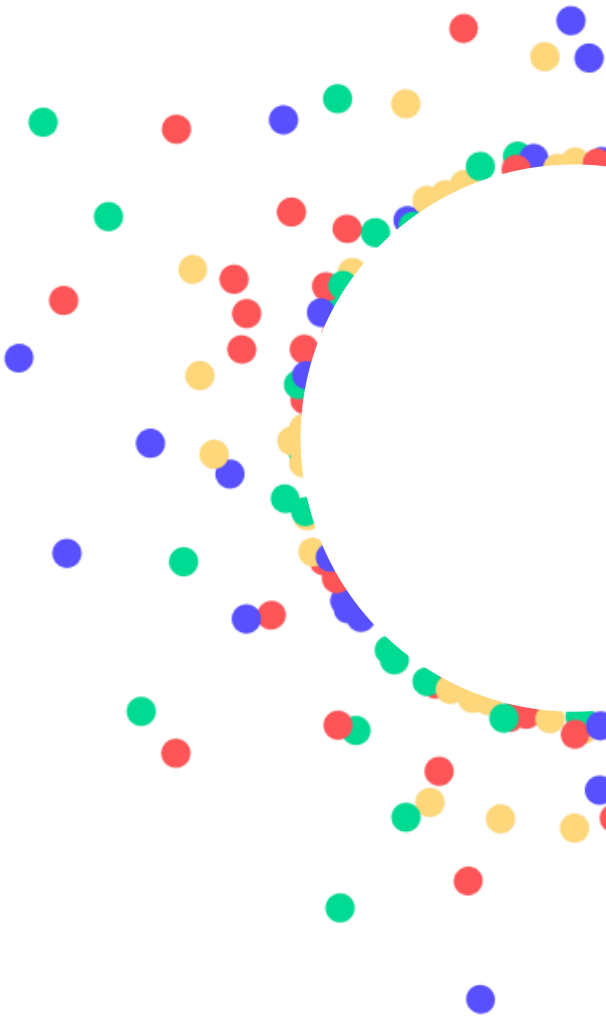
The world of monitoring has fundamentally changed. Today's monitoring tools were not designed for the complex, dynamic, and interconnected nature of cloud-native architecture. Companies need a monitoring solution that is as scalable, reliable, and flexible as the cloud-native apps they need to monitor.

Prometheus' single binary implementation for ingestion, storage, and querying makes it ideal as a light-weight metrics and monitoring solution with quick time to value — perfect for cloud-native environments. But simplicity and ease has its trade-offs: as organizations inevitably scale up their infrastructure footprint and number of microservices, you need to stand up multiple Prometheus instances, which requires significant management overhead.

Because of the time and challenges involved with running your own Prometheus environment, many organizations are exploring moving to a hosted, or a managed metrics and monitoring SaaS offering. To ensure a smooth transition to the solution and avoid future lock-in, it's critical that the SaaS monitoring solution be fully Prometheus-native. This whitepaper explores the key capabilities that organizations need to consider when selecting a Prometheus-native monitoring solution.

Key questions to ask your vendor

- What is the vendor's guaranteed uptime? How do they handle potential noisy-neighbor situations?
- What mechanisms does the vendor offer for keeping data growth, cardinality, and costs under control?
- How much customer time investment does it take to operate the vendor's tool? What components are run by the customer versus the vendor?
- Is the solution an instance of managed Prometheus? Or is it a SaaS solution that is fully Prometheus-native?
- Can the vendor host the SaaS solution in a different region or cloud than my production environment?
- How big is your largest hosted customer (unique time series over a period of time, ingested metrics/sec)?
- What is the total cost of ownership for running the vendor's solution? How do costs scale as you grow?



Overall vendor requirements

Before diving into the feature and functionality requirements of the solution you choose, it's critical to step back and assess the vendor and solution as a whole. There are several areas that you should consider, including:

Complete end-to-end managed solution

Even though they are SaaS offerings, several of the solutions on the market are not complete end-to-end offerings. For example, in some cases, you are still responsible for running your own instance of Grafana for dashboarding and visualization and Prometheus Alert manager for alerts. Other solutions also force you to continue running Prometheus collection instances in your own environment. This additional management overhead can ultimately prove to be very time consuming and expensive for organizations. That's why in many cases it makes more sense to work with a SaaS solution that is fully Prometheus compliant, instead of pure managed Prometheus. That will eliminate the need for additional tooling you run yourself

What does Prometheus-native mean?

A Prometheus-native SaaS solution must provide 100% support for:

- Prometheus ingestion protocols
- PromQL queries
- Prometheus Alert Manager definitions
- Prometheus rollup rules
- Grafana dashboards

Industry expertise and knowledge

Running large, or even mid-sized monitoring environments for cloud-native architectures takes a specific set of skills and expertise. You'll want to consider the years of experience, reference customers, and track record of the vendor and the team you'll be partnering with. One indicator of industry expertise is to look at contributions to open source tooling in the ecosystem.

Key questions to ask your vendor

- Is the vendor offering a completely managed solution? If not, what components is the customer expected to run?
- Tell me about the experience of the team running the SaaS solution? What other large-scale SaaS and cloud-native monitoring solutions have they run before?

Vendor focus on the metrics solution

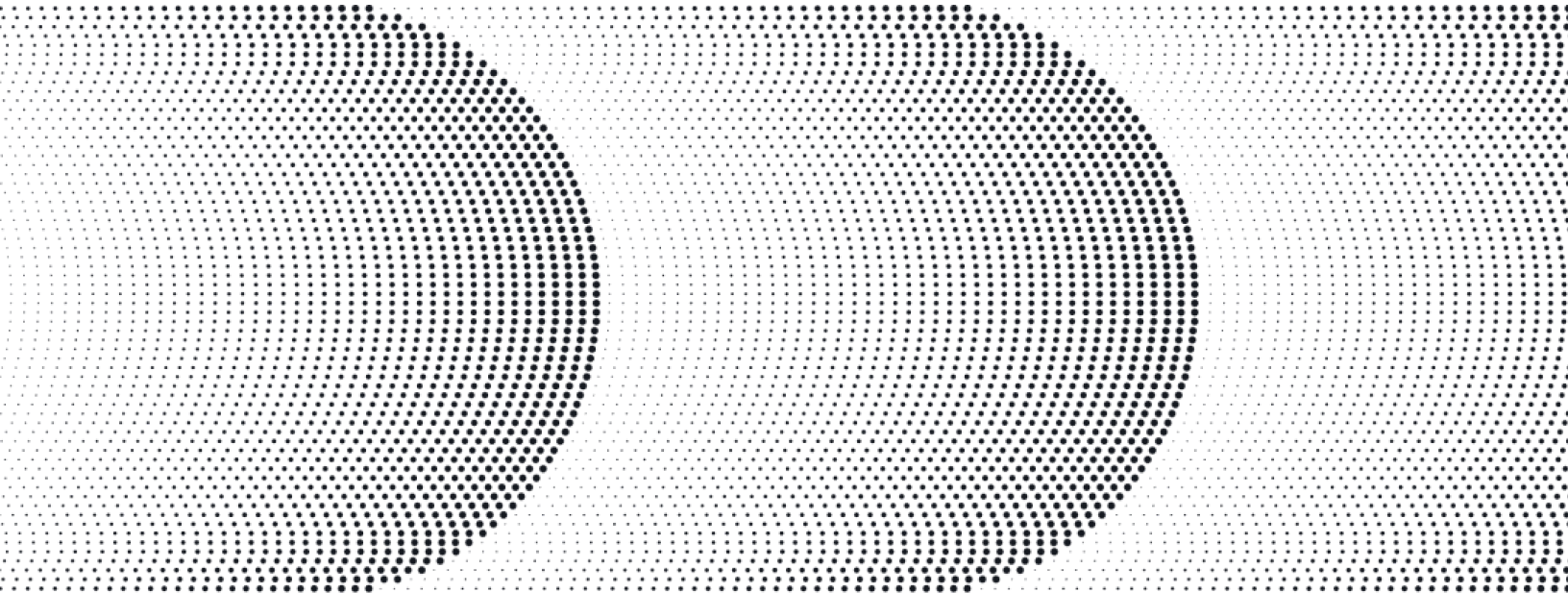
In a dynamic market with many new observability solutions entering, it can be easy for vendors to get distracted or spread too thin across many different products. You'll want to understand the vendors' long-term commitment to the Prometheus-native monitoring offering, and whether they are likely to have resources siphoned off to other projects.

Speed of innovation

Beyond focus, you'll want to assess how quickly the vendor is releasing enhancements to their Prometheus-native monitoring service. If it's a focus area for them, they should be releasing customer-facing enhancements every few weeks, if not continuously.

Key questions to ask your vendor

- What other offerings (SaaS or on-premises) do you offer in your portfolio? How many resources do you have dedicated to the Prometheus-native monitoring offering?
- How often do you release new or updated product enhancements above and beyond enhancements inherited from OSS?

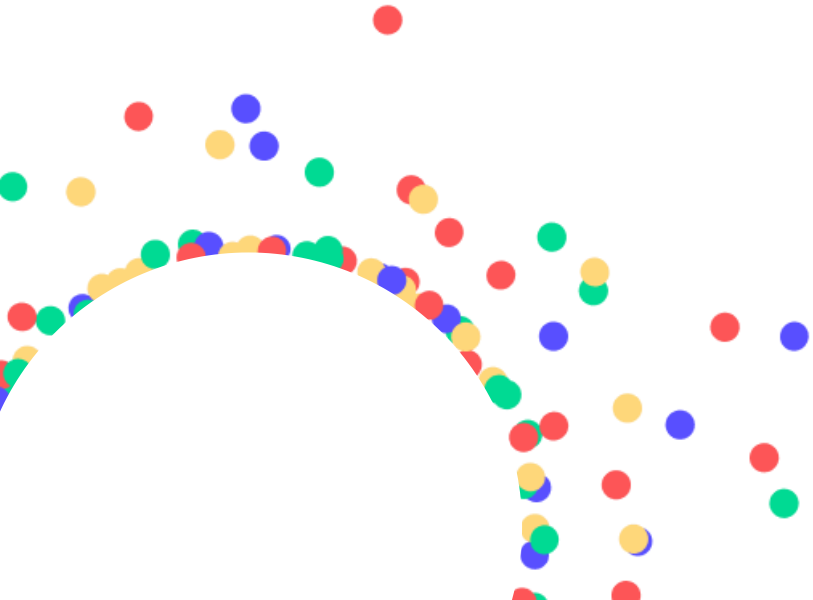


Functional requirements

Organizations who are considering a Prometheus-native monitoring solution should consider the following criteria as you make your short list.

Control over limits and retention

Cloud-native environments emit a massive amount of monitoring data — especially as developers add more metadata to their metrics causing massive cardinality spikes. As monitoring data volumes grow, so do costs. To combat data growth (and ultimately costs) organizations must make decisions based on business value about what data is kept, for how long, and at what resolution. This not only helps keep cost under control, but can make it easier and faster to find the data needed to solve problems. For example, a company may decide to keep production data for six months at a 30 second resolution and then for an additional two years at a 1 hour resolution. For non-production data, they may want a 15 second resolution, but only for two weeks. The first step to being able to make these decisions and trade-offs is the ability to see what teams and services are generating the most metrics with the highest degrees of cardinality. Once you have that visibility, you can then take action by either aggregating and rolling up metrics, or taking it one step further and setting quotas and limits by team to make sure no team is consuming more than their fair share of metrics.



Key questions to ask your vendor

- What capabilities do you provide to help with data growth and cost control? Can they help ensure you aren't ingesting duplicate metrics?
- Do you offer rate limiting? Can it be applied by team or label?
- Do you offer the ability to dynamically adjust the resolution of data?
- What aggregation and downsampling levers do you offer? At what point in the data stream does this occur?
- Does your solution provide a way to show which teams are generating the greatest volume of metrics and cardinality?
- What kind of cost-attribution metrics are available? Can you use the cost-attribution data to create granular alerts, customized dashboards or just export the raw metrics?
- What retention policies can you set in the product?

High availability and reliability

Under the strain of increased data volume and cardinality, monitoring systems become unreliable: they lose data or experience downtime. Without monitoring, teams are flying blind and won't be able to respond to issues in real time. It's important to consider where your SaaS monitoring is hosted so you don't accidentally locate it in the same region/cloud as your production environment and risk a simultaneous outage. Another important consideration is noisy neighbors, which is when another tenant in a multi-tenanted SaaS environment impacts performance of other tenants. Most vendors will offer a rebate or penalty paid if they violate an SLA, but what's more important is that they are doing everything possible to ensure the SLAs aren't violated in the first place.

Key questions to ask your vendor

- What is the vendor's uptime guarantee?
- What is the vendor's actual delivered SLA over the past 12 months for all customers? What is the vendor's actual delivered SLA for the top 10% of customers?
- What region and cloud is your SaaS offering hosted in? Can I request a different region or cloud?
- Do you offer single tenancy and multi-tenancy SaaS offerings?
- If multi-tenant, how do you protect against "noisy neighbors"?

Performance and scale

It's critical that you have a monitoring solution that you trust can scale reliably as you transition to cloud-native and continue to achieve high levels of performance. Under the strain of increased data volume and cardinality, some monitoring systems become unreliable: they lose data or experience downtime. Without monitoring, teams are flying blind and won't be able to respond to issues in real time. Losing data results in teams dealing with gaps in their metrics that make troubleshooting incidents more difficult. Even if the situation is not as extreme as unplanned downtime or data loss, many SaaS solutions suffer from performance degradation due to noisy neighbors if they are running a multi-tenant environment.

The other component to this criteria is scale: if you suddenly quadruple a large volume of metrics, there should be minimal operational input from your team.

Key questions to ask your vendor

- How many writes per second does your largest SaaS monitoring customer consume?
- How many active timeseries does your largest customer have?
- At what frequency are alerts checked? How fast are alerts generated?
- Do you offer any type of alert grouping or correlation?
- How many concurrent alerts do you support at your largest customer?

Security and administration

The last of the functional requirements is security and administration. You'll need to understand what fine-grained security and access controls the vendor has put in place to protect your data. You'll want to ensure that all user or machine communication is encrypted and that both user and service accounts leverage secure authentication systems. One of the best indicators of this is to look for adherence to compliance standards like SOC2 Type II.

Administration and access control is also a vital piece of the security puzzle. For example, developers must be able to see other teams' environments, but should not be able to make potentially breaking changes. Administrators should be able to set user permissions based on the scope of their role and responsibilities.

Key questions to ask your vendor

- Is all user or machine communication encrypted? What encryption protocols?
- How is user access management and role-based access control handled? Can users be granted granular permissions and access based on their role?
- Do you integrate with SSO/SAML authentication tools to automatically provision/deprovision users?
- How are service accounts authenticated?
- What infrastructure-level security standards are in place?
- Are you certified to any compliance standards, such as SOC2 Type II?

Service, support, and pricing requirements

While the functional requirements are the core focus area for buyers, services, support, and pricing cannot be overlooked. This area can often be a game changer, as many organizations require strict SLAs or need a dedicated team to help them ensure success. Look for a vendor who is a trusted partner — who can bring expertise to the table and has the skills to deal with the unexpected. What services beyond the product does the vendor provide: On-boarding of historical data? Assistance setting up alerts and dashboards? Enablement sessions with users and administrators? You'll want a vendor that does all of these things, AND has a pricing model that is fair and predictable and only grows as your value from the tool grows.

Key questions to ask your vendor

- How predictable is your pricing model? Is there a possibility of getting an overage?
- Is there an additional per-query charge on top of ingest and storage fees?
- Does the vendor offer on-boarding of historical data?
- What on-boarding services are available? Assistance setting up alerts and dashboards? Enablement sessions with users and administrators?
- How many years of experience does the technical support and account management team have?

Next steps: the bake-off

After you've completed a paper evaluation, the next step is to do a bake-off: Most vendors offer a free or paid pilot where you can test the capabilities and make sure it will meet your needs. Make sure you go into the pilot with a clear set of success criteria and a plan for how you will put the product through its paces.

As you go through this process, you may come to the conclusion that a pure managed Prometheus offering doesn't actually meet your criteria — instead what you need is a Prometheus-native SaaS solution. Chronosphere is the only SaaS monitoring solution built for cloud-native, providing deep insights into every layer of your stack — from the infrastructure to the applications to the business. Chronosphere is open-source compliant, supporting all major open source metrics ingest protocols, dashboards, and query languages. With Chronosphere, not only can teams avoid lock-in, but they can also leverage their existing Prometheus and Grafana investments. Additionally, Chronosphere supports older generations of metrics protocols (Graphite, StatsD, etc), meaning it will support your entire environment, even as you migrate off older formats.

Learn more and request a demo at chronosphere.io.