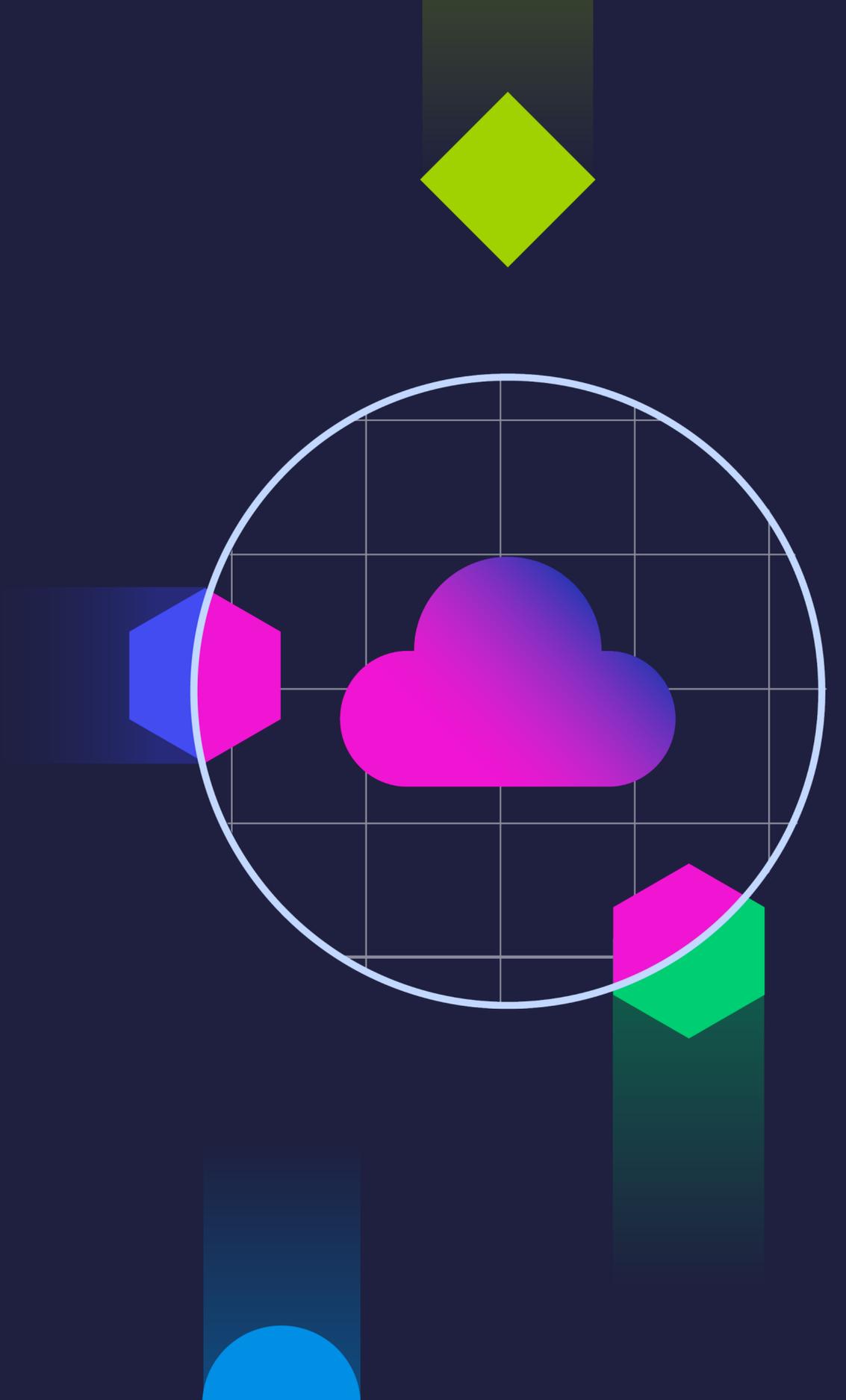
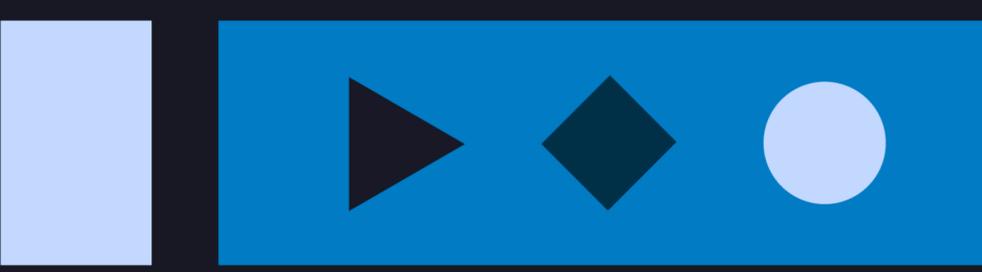
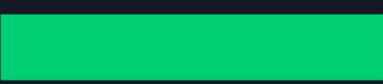
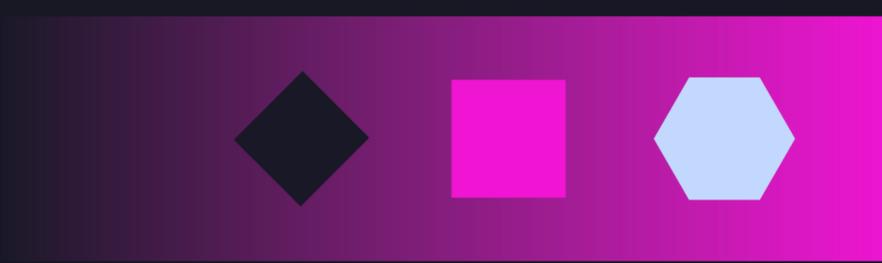


Should You Self-Host or Use Temporal Cloud?

A Technical Guide to Aligning Temporal with Your Needs





Whether you're considering self-hosting Temporal or adopting Temporal Cloud, this guide is designed to help you make an informed decision.

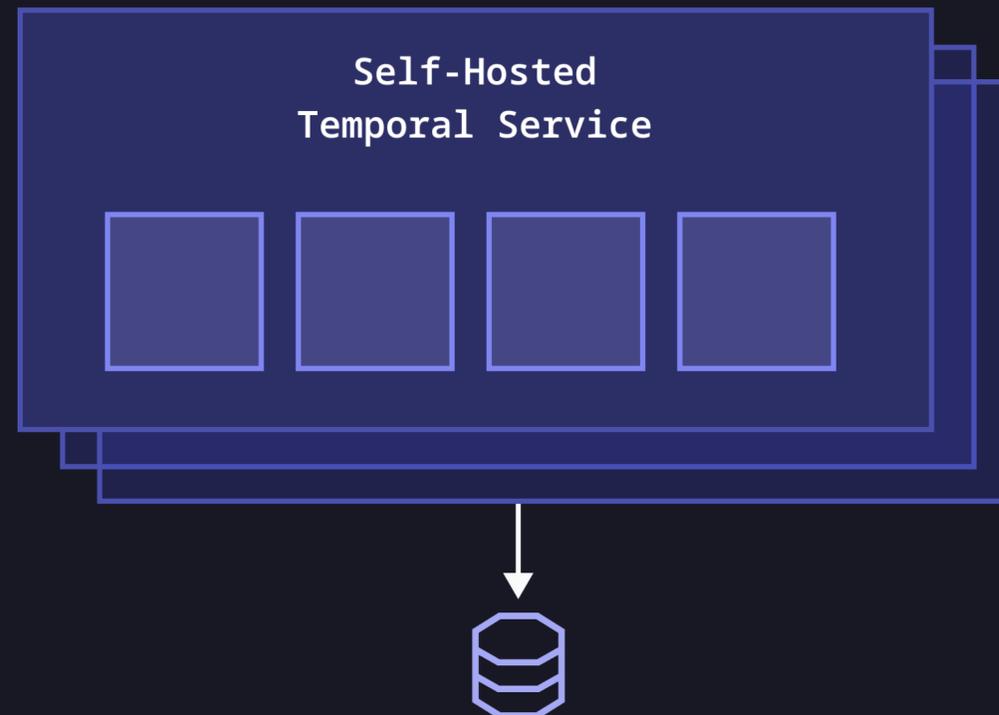
Temporal enables reliable, durable execution of workflows at scale. But as you look into your deployment options, it's important to understand how each approach aligns with your company's unique needs — right now and as you grow.

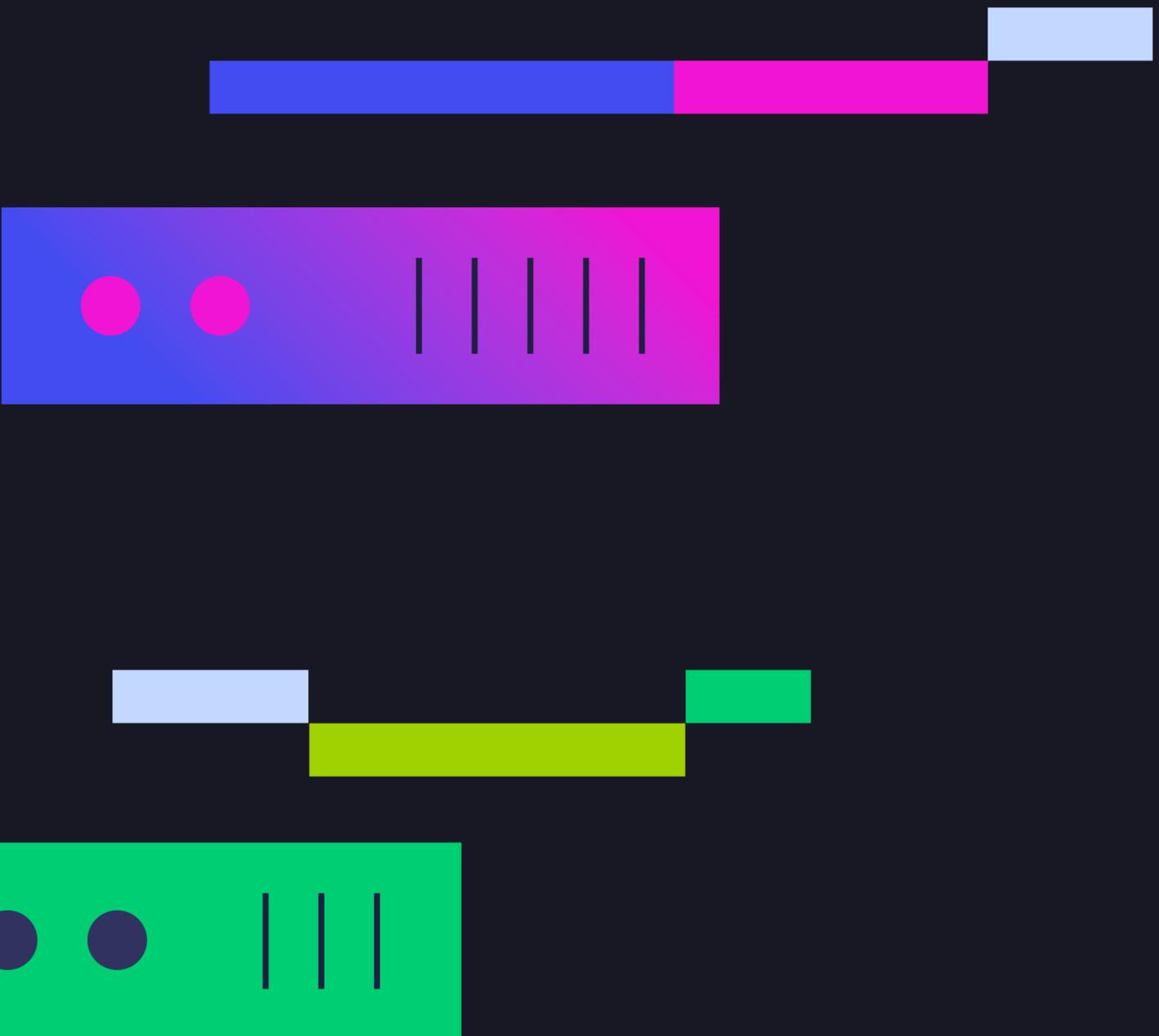
If your team is already self-hosting Temporal and you're wondering whether it's time to migrate to Cloud, or if you're just getting started and aren't sure which path to choose, this guide will help you make a sound decision.

Read on for insight that will help you navigate the decision between self-hosting and Temporal Cloud's "hands-off" services, and how to align these options with your needs.

Self-Hosting Temporal

Temporal is built on a foundation of open source, supported by a permissive license that empowers you to adapt and scale it to meet your unique needs. Whether you're just getting started, meeting specific on-premises requirements, or exploring Temporal's potential, self-hosting offers the flexibility to innovate on your terms while aligning with your technical and operational goals.





Self-hosting Temporal is a good fit for organizations that:

- ✓ Require on-premises deployment to meet regulatory or operational constraints.
- ✓ Cannot depend on a cloud service due to organizational or legal restrictions.
- ✓ Operate in GovCloud environments, where specialized compliance and security requirements apply.
- ✓ Have a strong infrastructure team capable of managing databases, Elasticsearch, observability tooling, and other essential components at scale.
- ✓ Handle consistent, slow-changing workloads, where the predictability of resource requirements reduces operational complexity.

While self-hosting provides full control and customization, it comes with significant responsibilities and challenges. Ensuring your Temporal instance is reliable and scalable requires careful planning and ongoing investment in infrastructure, deployment, and staffing. These costs and complexities should be weighed against the benefits of flexibility and independence.

Staffing

Many customers may start with a single senior engineer to deploy and operate Temporal. However, as Temporal becomes business-critical infrastructure, a single engineer may struggle with:

- The demands of running a highly available, secure, and stable service.
- Supporting developers using Temporal.
- Dealing with operational issues

While this approach may initially feel more affordable, it also poses the risk of relying heavily on a single individual.

Two common options to reduce this risk and increase coverage include splitting the work between two more senior engineers or building a team of three junior engineers.

This collaborative approach reduces risk and increases on-call coverage. For many organizations, the cost of maintaining a dedicated team of platform engineers for a self-hosted Temporal instance often becomes a significant consideration. At this scale, moving to Temporal Cloud offers a more cost-effective solution, reducing operational overhead and enabling teams to focus on innovation.

With Temporal Cloud, your team can reallocate resources from maintaining infrastructure to focusing on innovation, making it a cost-effective and scalable alternative as your organization grows.

Additionally, choosing to self-host requires significant engineering and ongoing effort to resolve several potential challenges:

- Security and compliance
- Scalability for spiky or growing workloads.
- Multi-region hosting and low-latency performance.
- Uptime, availability, and reliability challenges.
- Complexities in managing control planes, audits, and user access.
- Ongoing maintenance, upgrades, and expert support.

Temporal delivers [durable execution](#) through its architecture, including state persistence, retry policies, and timeout handling. However, the responsibility for ensuring the infrastructure can reliably support this execution falls on the self-hosting team. For example, improperly configured or insufficiently scaled databases and inefficiently configured Temporal Servers are common issues that can impact the service's ability to function effectively and meet your operational requirements.



ne

#RUBY - SDK

#SUPPORT - COMMUNITY

#TYPESCRIPT - SDK



ne

These infrastructure considerations are critical for maintaining the reliability of a mission-critical Temporal Service. A mismanaged environment can lead to downtime, which risks significant customer impact and undermines the benefits of Temporal's durable execution capabilities.

That said, we see major customers successfully self-hosting Temporal for their critical workloads. In nearly all such cases, they have dedicated teams operating Temporal 24x7, supported by mature observability and alerting infrastructure. For organizations equipped with the resources and expertise to manage these complexities, self-hosting can be an effective option to retain control and customize their deployment.

Infrastructure

When it comes to operating and resourcing self-hosted Temporal for critical production workloads, many Temporal customers actively share their strategies and best practices within our community.

Our community regularly collaborates to provide practical advice on efficiently [scaling](#) Temporal in production environments across use cases. Our [Slack community](#) of over 14,000 members, including Temporal experts, is the perfect place to learn and share strategies for optimizing deployment costs, overcoming scaling challenges, and fine-tuning resource allocation. Whether you're seeking advice or exploring best practices, there's always someone ready to help.

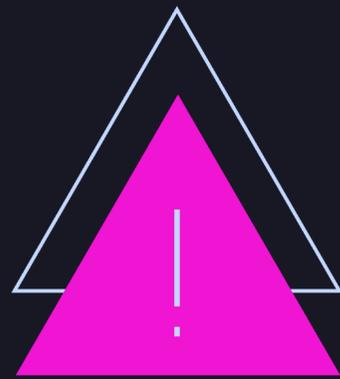
“Temporal is renowned for its flexibility and robustness, but mastering it in a self-hosted environment requires specialized knowledge and strategies, and we would like to share our learnings.”

Varun Gupta

VP of Software

Engineering at Salesforce

Maintenance

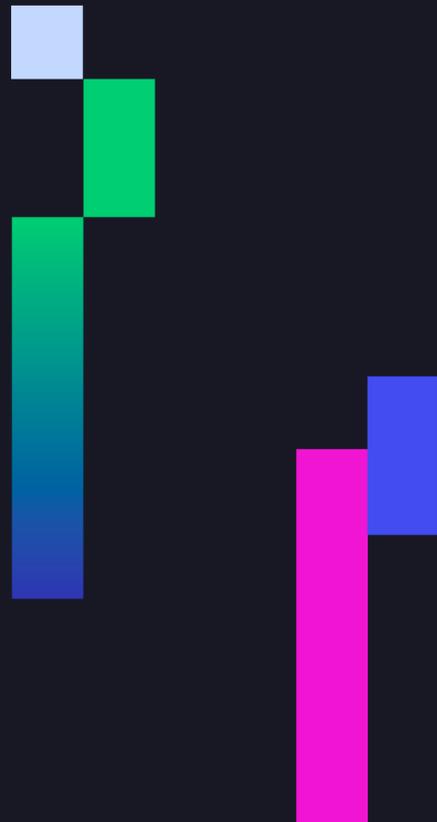


While planning, triaging, and recovering from incidents is a part of operating any production system, self-hosting Temporal places the entire responsibility for these tasks including infrastructure management, monitoring, and recovery protocols on your team. In contrast, Temporal Cloud minimizes this burden by managing these critical operational aspects for you

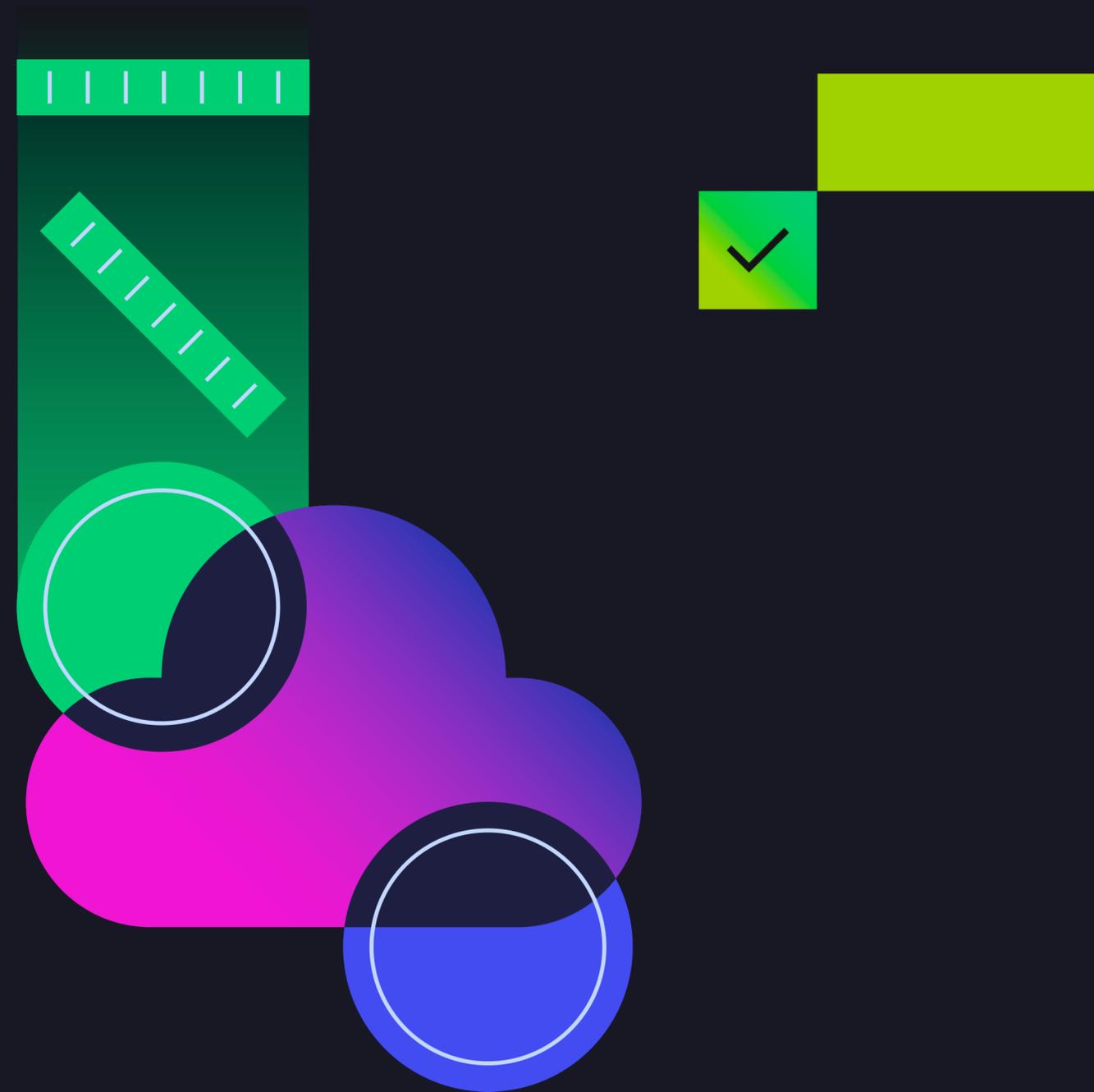
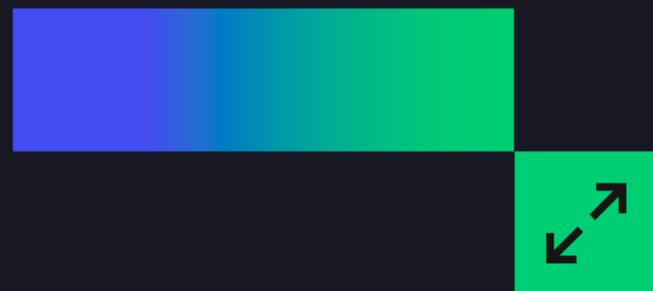
You'll need to be prepared for:

- System upgrades and patching
- Scaling and migrations
- Proactive monitoring to avoid downtime
- Incident response and triage

In contrast, Temporal Cloud minimizes this burden by managing these critical operational aspects for you. While these self-hosting responsibilities allow for deep control and customization, they can also introduce complexity and unanticipated costs.



Reducing Cost and Scaling with Temporal Cloud



Feature Comparison

Most Temporal capabilities are available in both the open-source (OSS) and Temporal Cloud deployments, ensuring a robust and flexible foundation for your workflow management needs. However, there are key considerations to keep in mind when choosing between self-hosting and using Temporal Cloud.

The table to the right highlights the essential features of each option, helping you evaluate which deployment aligns best with your technical requirements, operational priorities, and scalability goals.

	Self-Hosted (OSS)	Temporal Cloud
RBAC	Available with a custom plug-in	Built-in
SSO, SAML, SCIM	Requires custom integrations	Included in Enterprise & Mission Critical
Compliance	SOC 2 or HIPAA compliance requires internal implementation	Comes with SOC 2 Type II and HIPAA compliance
Private Link	Manual setup	Managed
Service Accounts, API Keys	Custom implementation	Managed
Audit Logging	Custom implementation	Centralized and built-in
Cloud availability (AWS, GCP)	Manual configuration	Seamless integrations
MRN Support with Replicated Namespaces	Requires complex setup	Fully managed
On-Call Support	Your own support team	24x7 support available

“I trust Temporal to operate the technology better and more efficiently than we ever will be able to do—even if we fully staffed an internal team to run it.”

Rob Zienert

Senior Software Engineer
at Netflix

Most Temporal capabilities are available in both the open-source (OSS) and Temporal Cloud deployments, ensuring a robust and flexible foundation for your workflow management needs. However, there are key considerations to keep in mind when choosing between self-hosting and using Temporal Cloud. The table to the right highlights the essential features of each option, helping you evaluate which deployment aligns best with your technical requirements, operational priorities, and scalability goals. Temporal Cloud is built on the same open-source foundation as self-

hosted Temporal, ensuring reverse compatibility. This means you can easily transition to self-hosting if your needs change, giving you flexibility without vendor lock-in.

As your usage of Temporal grows, the complexity of managing larger-scale deployments increases. What may begin as one or two production use cases can quickly expand to a broader range of workflows with varying requirements, each demanding unique setup, tuning, and configuration.

Even for customers operating within a single region, scaling to support diverse use cases and workloads introduces challenges in maintaining efficiency, reliability, and performance.

Temporal Cloud helps you navigate this complexity by providing a fully managed service that scales with your business. This way, you can focus on building workflows rather than managing infrastructure. Here’s how Temporal Cloud simplifies your infrastructure management and allows you to scale without difficulty.

Scalability

When self-hosting Temporal, you're faced with the challenge of planning for infrastructure to support unexpected workloads, including managing history shard counts and ensuring high availability.

For example, Temporal's history shard count can't be increased after creation. This means you often have to overprovision for your average workload to ensure [scalability](#) during peak demand.

With Temporal Cloud, you eliminate these constraints. Temporal Cloud automatically scales based on demand, removing the need for manual capacity adjustments.

Temporal Cloud automatically scales your workflow capacity based on demand. Each Namespace has a default rate limit of Actions per second (APS), which adjusts according to the previous week's usage. This ensures continuous, hands-free scaling that can handle varying workloads without ever dropping below the initial APS threshold.

Temporal Cloud's automated scaling ensures that your infrastructure grows in line with your usage, and additional capacity can be requested seamlessly, without downtime or disruption to your workflows.

[Bugcrowd's](#) team sums up their ease with scaling like this, "Thanks to Temporal, [we] can launch services twice as fast, with 400% more capacity and 50% less downtime."

As Bugcrowd experienced, Temporal Cloud helps to avoid the challenges of manual capacity planning to free up time for faster deployments and greater innovation.

With Temporal, Bugcrowd launches services twice as fast and experiences

400%
more capacity

50%
less downtime

Unified Control

Self-hosted Temporal requires complex operational management. To run a production service, you may need full-time DevOps for a Cassandra database backend, as well as for the Temporal Service itself. For this, we provide [helm charts](#) to aid with Kubernetes deployments.

While there are resources for you to reference to perform this work, the setup itself adds another layer of complexity and requires significant ongoing maintenance. Eventually, you will need to maintain local customizations and infrastructure, making this a difficult and resource-intensive task.

Temporal Cloud solves this problem by offering a unified view of all your namespaces across regions. It allows you to manage your entire deployment from a single control plane, simplifying operations and increasing efficiency. Temporal Cloud also includes built-in external observability, enabling you to monitor workflow performance, trace execution paths, and gain deeper insights without additional setup. Additionally, Temporal Cloud supports RBAC functionality, enabling you to delegate responsibilities across teams with ease.

This centralized control means you no longer need to add another layer of infrastructure or manage complex data flow decisions. Temporal Cloud handles it for you, saving both time and effort while ensuring your workflows are consistently managed across your entire organization.

As Josh Wickham, Principal Software Engineer at [Turo](#), explains, “Temporal offers a novel approach to decomposing our monolith, which aligns perfectly with our goal of transitioning to a microservices architecture.”

With Temporal Cloud, the Turo team was able to rely on Temporal to simplify their operations and reduce infrastructure complexity, and you can do the same.

Security and Performance

Unlike self-hosted Temporal, which requires manual configuration and custom plugins for security and compliance, Temporal Cloud simplifies many aspects of security with features like RBAC, SSO support, and audit logging. However, for data encryption, both self-hosted and Temporal Cloud deployments necessitate implementing a custom Data Converter and hosting your own Codec Server. The key difference is that in Temporal Cloud, you can set encryption configurations at the namespace level, offering more granular control compared to the server or session level settings in self-hosted setups.

Additionally, Temporal Cloud undergoes third-party penetration testing and is SOC 2 Type II and HIPAA compliant, providing assurance that your workflows are secure and compliant.

Beyond security, Temporal Cloud provides high-performance infrastructure with consistent low-latency. Temporal Cloud delivers a p99 latency SLO of 200ms per region, ensuring that high-demand workflows perform reliably, even at scale. The latency benchmarks on the right show you the Temporal Cloud's ability to support demanding workflows without the added operational complexity of self-hosting.

Operation	p90	p99
StartWorkflowExecution	24ms	54ms
SignalWorkflowExecution	14ms	40ms
SignalWithStartWorkflowExecution	24ms	61ms



The Cost Efficiency of Temporal Cloud

Temporal Cloud delivers significant cost savings and flexibility by eliminating the overhead of maintaining infrastructure. Our consumption-based model ensures your costs grow only with your actual usage, allowing you to [scale](#) efficiently as your needs change. Additionally, Temporal Cloud offers flexible support plans, allowing you to adjust your support level to align with your evolving operational needs and budget, further optimizing your total cost of ownership.

✓ Pay for what you use

Temporal Cloud remains true to its usage-based model, charging only for the [Actions](#) and Storage your applications consume. Bundled allocations in each plan cover initial usage, so you avoid overprovisioning or paying for unused capacity.

✓ Scale consumption, not hardware

Temporal Cloud not only incorporates the best practices we have amassed through thousands of deployments, but it also allows you to scale smoothly and only when you need to, so you don't need to plan and acquire new infrastructure to handle growth.

✓ Spikey traffic, smooth scale

Unlike traditional setups, Temporal Cloud can handle traffic spikes without requiring you to provision extra hardware for peak times. You only pay for what you use, which means it's great for fluctuating workloads.

✓ Volume discounts that grow with you

Volume pricing now starts at just 5 million Actions (down from 300 million), with discounts applied across all namespaces in an account. This holistic approach rewards your growth, delivering even better pricing as you scale without requiring a credit purchase.

✓ Tailored plans for every need

Whether you're a small team just starting out or an enterprise running mission-critical workloads, Temporal Cloud's plans offer the right combination of features and support. For example, Essentials provides affordable access for lower usage needs, while Mission Critical includes 24x7 support and dedicated assistance for stringent uptime requirements.

Take the Best Next Step

Whether you're considering self-hosting or moving to Temporal Cloud, we recommend speaking with one of our Solutions Architects or Business Partners to explore the best approach for your specific needs.

Our Solutions Architecture team is available to help you design and optimize your use cases. We also offer ongoing professional services for hands-on support throughout your implementation.

Deploy your Temporal Service today with confidence.

Learn more from our docs and sign up for Temporal Cloud and experience hassle-free, scalable infrastructure from day one.



Get started with Temporal
SET UP YOUR LOCAL ENVIRONMENT NOW

See how it works in your language
WITH EXAMPLE APPLICATIONS

