

**Cloud's future
runs through
Sovereign DBaaS**

several9s

Table of contents

1

Enter Sovereign DBaaS

2

Key takeaways and recommendations

3

Background and definition

4

Key characteristics and features

5

Benefits and challenges

6

Sovereign DBaaS vendors

Cloud computing is a generational shift that has transformed how modern solutions are being developed and deployed. Database-as-a-service (DBaaS) has witnessed an enviable and inexorable growth in recent years. Yet, we are still in the early stages. A vast amount of legacy data and analytics systems are still waiting to be modernized.

However, cloud computing is not just “public cloud computing.” The deployment models do not necessarily belong only in a hyperscale cloud provider’s data center but may also be on-premises or managed service providers. Savvy IT leaders are looking to design their system architectures to leverage the most cost-efficient and secure deployment models and are seeking flexibility to redeploy relevant workloads in the most optimal location and with the least amount of friction. Often this includes deploying open source databases across multiple locations to avoid vendor lock-in and reduce cost.

Public cloud has spurred innovation at an unprecedented rate. It has shifted the focus from organizations spending valuable resources on IT infrastructure to solving business problems with minimal overhead and faster. But, not all workloads can partake in its advantages.

Certain workloads are beset with rising costs and complexities, especially in ensuring data security and complying with increasing global data privacy regulations — data is subject to the laws and regulations of the countries in which it is collected. However, once shared, it is complicated to track how the data is used in regions with their own laws, which may not be as strict as those of the origin.

1

Enter Sovereign DBaaS

Enter Sovereign DBaaS

Sovereign DBaaS provides organizations full control over where their data is stored and how it is managed. With this level of control, other benefits accrue, such as predictable IT spend, transparency into security configurations, and the ability to meet regulatory compliance, such as the EU GDPR and the CCPA.

Through the use of open-source DBMS, Sovereign DBaaS can also lead to lower costs and portability of workloads across hybrid multi-cloud environments. To achieve these benefits, the vendor must provide a hybrid multi-cloud management and automation console.

Sovereign DBaaS gives ultimate deployment portability with no vendor or environment lock-in and full control over costs, security, and configuration, achieving the promise of open-source databases through a vendor-neutral distribution.

Sovereign DBaaS can also be an on-ramp to public DBaaS for organizations still maturing their public cloud strategy and have not yet selected their future cloud service provider.

This whitepaper explores the rationale and benefits of using Sovereign DBaaS. It establishes a common understanding of the term and examines its key characteristics and appropriate use.

2

Key takeaways and recommendations

Key takeaways

- **Sovereign DBaaS relies** on open-source/source-available databases and gives the customer full control over its configuration and deployment options. Convenience rather than cost is the main driver for early cloud adoption, but the cost of expansion is unknown when there are many variables. Sovereign DBaaS can help reduce costs and improve its predictability.
- **Sovereign DBaaS enables** cloud vendor-neutral and environment-agnostic deployment of open-source databases with complete control. The vendor provides a management console to manage and migrate workloads between on-premises and multiple cloud providers.
- **Workloads with highly sensitive data** requiring compliance with data privacy regulations can achieve data residency goals by using Sovereign DBaaS. It helps organizations distribute sensitive and nonsensitive workloads across different deployment options but manage them through a single pane of glass control plane or a management console.
- **Sovereign DBaaS delivers** a predictable quality of service. Use cases with strict latency and throughput performance needs benefit from the flexibility it affords.

Recommendations

Data leaders modernizing their data infrastructure to deliver the most cost-efficient and secure data analytics architecture should:

- **Perform cost-value analysis** to determine which applications are best suited for public, private, or Sovereign DBaaS.
- **Assess open-source databases** that run across all deployment models as they can provide portability, lower total cost of ownership, and avoid cloud vendor lock-in.
- **Evaluate the latency needs** of different types of workloads to influence a future state architecture comprising multiple deployment models to meet cost-performance

and security goals.

3

Background and definition

Background

Recent public cloud outages have once again renewed interest in alternate options that can still leverage the ease of use of cloud deployments but also provide more control over the data infrastructure. In December 2021, AWS suffered three outages in two weeks. In early 2022, several significant data breaches rekindled the debate on the need to deploy hybrid multi-cloud architectures. Organizations that rushed head-first into the cloud are now reassessing their options.

Although public cloud providers have sophisticated infrastructure compared to the vast majority of organizations' own data centers, these incidents raise important questions concerning ways to mitigate downtime, security breaches, cost, and latency. Hence, it is imperative to revisit the rationale for multiple deployment models.

Figure 1 shows a graph that makes the case that public cloud, co-location, and on-premises deployments have specific advantages to the type of workload being executed. Ephemeral tasks, i.e., they run in bursty modes for short durations of time, are best suited for public cloud, while workloads that require consistent use of resources for longer durations may still be most effective on-premises.

How can an organization leverage all three of these deployment options based on their specific workload characteristics in the most seamless manner? This is the promise of Sovereign DBaaS.

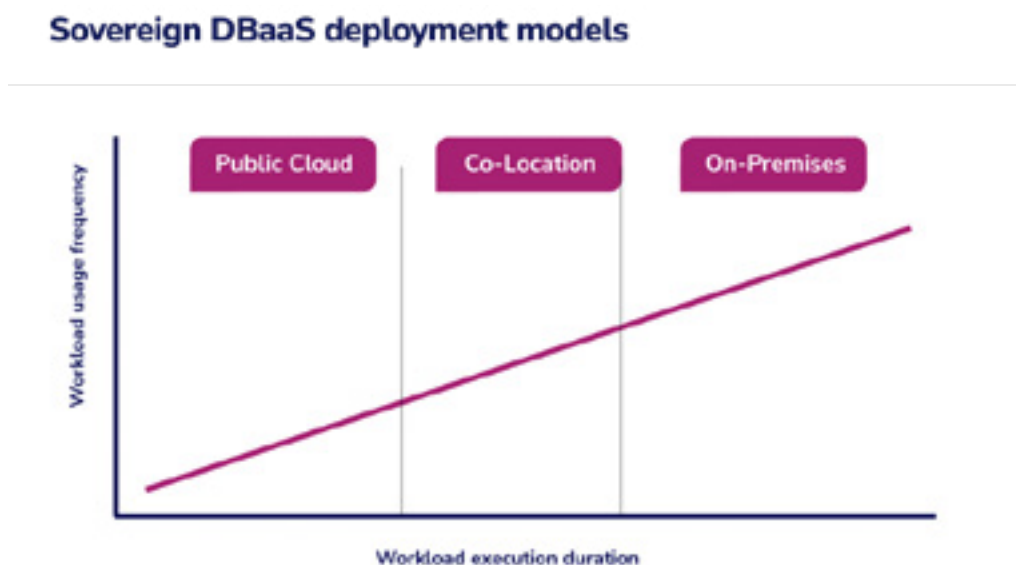


Figure 1. Tasks running for short bursts of time and do not have high sustained resource consumption are best suited to cloud's elasticity. All other tasks should evaluate co-location or on-premises deployments. Other factors that affect the use of Sovereign DBaaS are costs, data residency requirements, and latency.

To illustrate this concept, let's look at the use case of tax filing software, such as Intuit's TurboTax. In the USA, the tax filing deadline is April 15 (or the next working day if April 15 is on a weekend), when hundreds of millions of people file their taxes electronically. A significant number of filers do so in a few days leading up to the deadline, meaning that the underlying resources are stretched to an extreme scale for a very short period of time. Public cloud's on-demand, infinitely elastic infrastructure is most suitable.

Conversely, heavy sustained usage workloads are not able to take advantage of cloud's elasticity and may run more cost-effectively on-premises. Let's consider another real-life use case: the media streaming company, Netflix, renowned for being fully deployed in the cloud. However, a deeper examination shows that this is only true for their operational workloads. Its content—movies and shows—is deployed on its own appliances, called Open Connect, installed at various service providers' data centers.

So far, we have looked at two examples of applications with varying types of workloads. Now, let's look at the implications of the data storage layer. Most organizations are already beset with multiple copies of redundant data that lead to data silos and trust issues. As data volume and security concerns increase, we want to minimize unnecessarily copying and moving data.

Moving data from where it is generated to a cloud facility is not only expensive but incurs latency and necessitates governance. Similarly, moving the data out of the public cloud is also expensive despite egress costs going down. Sovereign DBaaS offers the hope that data can be kept in the place where it originated and managed in a single, governed, and consistent manner.

Definition

Let's face it; public DBaaS has taken off like hot cakes. Almost the entire growth of the database market is in the cloud, not on-premises. Meanwhile, Private DBaaS has caused confusion about the meaning of the term "private," which is often linked to simply on-premises deployment. But that was not the original intent; the fundamental characteristic of a private cloud is "tenancy," not its location. Simply put, a private cloud is dedicated to a single organization or purpose, whether in a public cloud or elsewhere. Sovereign DBaaS signals the "intent." It conveys the message that organizations need to have authority over their data to comply with the rules and regulations of the region where their data originated. It gives organizations complete control over the locations of their dedicated tenants in a hybrid multi-cloud environment. The location of the

infrastructure may be hosted on-premises, in a colocation facility, a managed service provider, or in a hyperscale cloud provider facility as infrastructure-as-a-service (IaaS).

Sovereign DBaaS is a service that provides access to a database management system in a dedicated tenant in any location that meets data privacy, latency, and cost goals.

Now that we have defined Sovereign DBaaS' meaning, the following section looks at its key characteristics.



Key characteristics and features

Key characteristics and features

Sovereign DBaaS is an evolutionary step over a single deployment model, leveraging public cloud's computing advantages with additional security and risk reduction benefits; figure 2 maps its features.

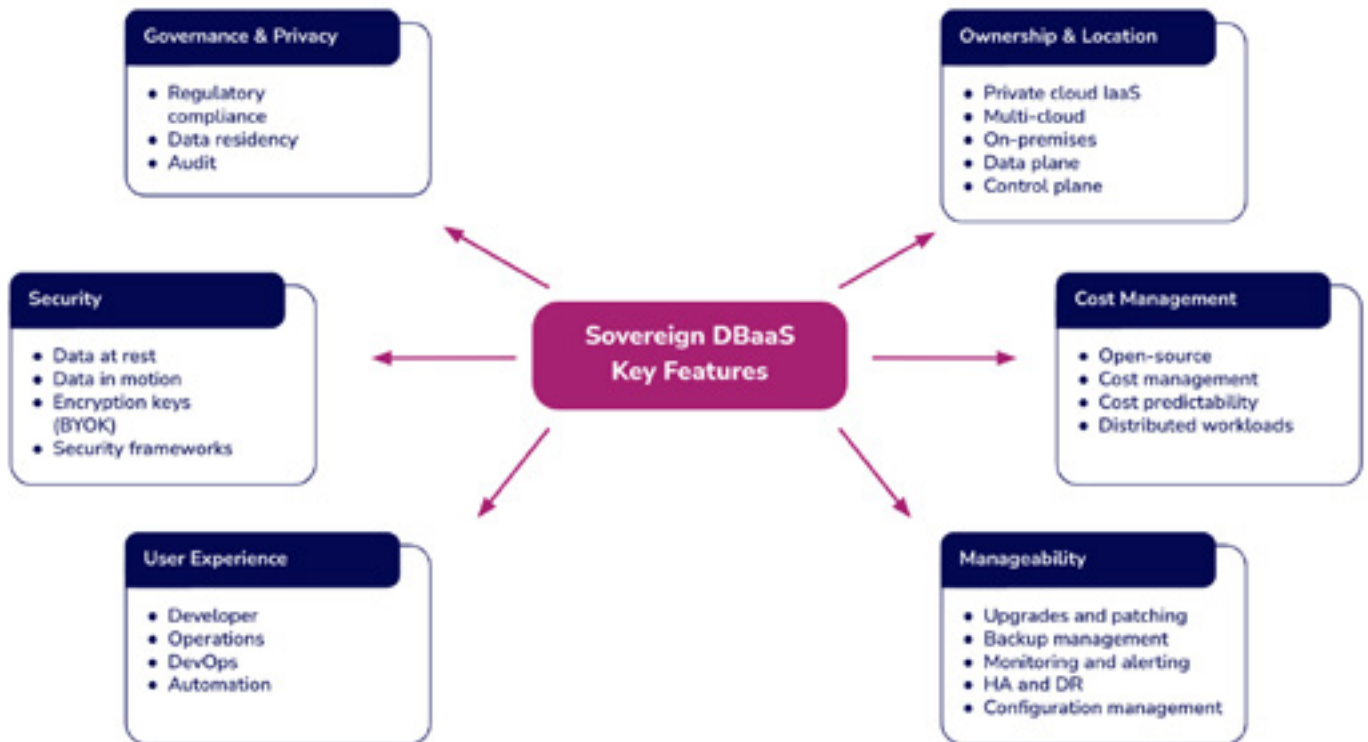


Figure 2. Sovereign DBaaS key characteristics

While public DBaaS is well understood and has become a commodity, Sovereign DBaaS is only now getting attention. It differs from its public counterpart in subtle manners, as this section explains.

Ownership & location

While the Sovereign DBaaS tenant deployment location can vary, it is mostly defined by the fact that the end-user controls the infrastructure. Even if the infrastructure is deployed inside a public cloud facility, the end-user has root access and the ability to install and manage any needed software components.

As users demand more control, even public cloud providers are responding by increasing their capabilities in this space. The recent launch of Amazon RDS Custom attests to this trend. Amazon RDS Custom allows customers to select the version number, update cadence, and ability to access the host operating system. However, this RDS offering must run in AWS and only supports proprietary databases like Oracle and SQL Server. Sovereign DBaaS' use of open-source databases gives it the fungibility of running in any cloud provider or within the organization's own or co-located data center.

Sovereign DBaaS infrastructure control should be examined across data and control planes as follows:

Data plane

It should be in the customers' own private cloud (data centers in case of on-premises, or VPC in AWS and GCP, and VNet in Azure). This provides customers full security as their data never leaves their own private storage. Many organizations have reserved capacity or committed credits. Compute charges can be offset against those reservations. Sovereign DBaaS integrates with the public cloud. For example, Severalnines provides the capability to stretch into cloud IaaS. However, implementation of "stretch" proprietary DBaaS databases, such as SQL Server Stretch Database, was unsuccessful.

Control plane

Control planes, or management consoles, may be in the same VPC as the data plane. However, many vendors prefer to offer the control plane inside their own VPC. This vendor VPC may not even be in the same cloud provider as your data plane, but its location is abstracted from the end-users. Some vendors allow customers to choose where they want to run the control plane as long as it can access the data plane. Control planes do not store your data but only the metadata needed to monitor and automate management tasks. You should evaluate its capabilities and functionality.

In summary, Sovereign DBaaS vendors do not own the hardware. The hardware comprises bare metal or virtual machines like VMware/KVM. These may exist in customers' own data centers, a co-location vendor, such as Equinix, or virtual machines from a cloud service provider, such as EC2 from AWS or Google Compute Engine from GCP. They may also use PaaS offerings, such as OpenStack or OpenShift.

Open-source & cost management

Rising costs have become one of the biggest concerns of public DBaaS. Public clouds' "fully-managed DBaaS" is typically three times the cost of bare metal DBaaS. Public cloud providers' PaaS services lower costs via multi-tenancy. However, dedicated tenants are not shared; hence raw costs may be higher.

As Sovereign DBaaS gives the users control over their deployment stack, it has the potential to lower costs and make them more predictable. Organizations perform a complete total cost of ownership (TCO) exercise to assess costs for public cloud PaaS versus Sovereign DBaaS.

Sovereign DBaaS helps in managing costs in the following manner:

- **Using open-source technologies, such as MySQL as DBaaS**
- **Simplifying and unifying cost management**
- **Improving cloud cost predictability and transparency and adding guardrails**
- **Distributing workloads across more cost-efficient hybrid multi-cloud deployment options**

A major cost-saving aspect is through the use of **open-source DBaaS**. In fact, they become a necessity when multiple clouds are scoped as most proprietary DBaaS solutions cannot run on others' clouds. Even open-source databases, such as MySQL and PostgreSQL, are deployed differently in each cloud provider, reducing their compatibility. In summary, Sovereign DBaaS achieves lowest cost and predictability through the use of open-source databases, although it is not just limited to open-source databases.

Simplified **cost management** is through a single pane of glass that abstracts and unifies each cloud provider or on-premises costs. Most cloud providers list granular costs that can run into many pages, thereby adding more complexity. In fact, many organizations staff a FinOps team whose job is to understand and reduce cloud costs.

Organizations achieve **cost predictability** for their software components by negotiating an enterprise license agreement (ELA), where they pay a fixed fee for a certain volume of database instances instead of incurring public cloud's variable costs. They can also better control high egress costs by deploying on-premises or in cloud providers with more favorable and predictable pricing policies. The egress cost is due to database replication, which becomes a cinch due to Sovereign DBaaS offering compatibility between hyperscale and smaller cloud providers.

To understand the **workload distribution** advantage, consider data scientists' preference to use GPUs to train models using deep learning techniques. For very large models, hundreds or thousands of GPUs may be used. However, GPUs cost considerably more than CPUs, making training of some models cost-prohibitive. The cost of GPUs is higher in the public cloud compared to the private cloud. But investing in on-premises GPUs may not provide cost benefits if the GPUs' usage is not sustained over an extended amortization period.

Hence, organizations must analyze each workload to determine the most cost-effective deployment option. Sovereign DBaaS gives you the freedom to pick the most optimal location for your workloads dynamically. This is possible through consistent user experience and a common single pane of glass management plane.

User experience

Sovereign DBaaS provides a consistent user experience regardless of the deployment location. User experience spans various producers, operators, and consumers of the data management pipeline. It should be examined across different personas:

✔ **Developer experience** (ease of development)

Developers should not have to use different tools or development environments (IDEs) when they work across different deployments. Having a common approach reduces development overhead, improves root cause analysis and collaboration, and reduces the time to development.

Sovereign DBaaS standardizes on the same versions of the DBaaS. This further removes the confusion of which extensions are enabled where and the differences in their capabilities. Developers can use the same APIs irrespective of where the DBaaS is deployed.

✔ **Operational experience** (ease of management)

Sovereign DBaaS' signature key characteristic is extreme configurability and customizability. This may require privileged access to the underlying resources. A single pane of glass unifies the management of all the databases handled by the Sovereign DBaaS—both open-source and proprietary. It also helps to standardize operations and replace manual tasks with automation.

User experience improves when deploying compatible software binaries across deployment locations. This experience can further benefit from standardization through a DevOps toolchain comprising open-source CI/CD tools, including Chef Puppet, Ansible, Jenkins, etc.

With the ability to assert full control over operational aspects, the administration team must guard against accidental (intentional or otherwise) actions that can impair the operations. This topic is covered in the next section on manageability.

Having a consistent user experience also helps in reducing costs as it reduces the cost of training and hiring.

Manageability

Digitalization and modernization projects are multi-year projects that involve regular experimentation and trials. Typical projects lock users into a homogeneous ecosystem. These organizations are reluctant to extend their learning curve and spread their wings into uncharted territories involving multiple deployment options. They may lack the skills or culture to embrace the attendant complexities.

Sovereign DBaaS can abstract the hybrid multi-cloud complexities and make the experience similar to their public DBaaS. This necessitates a single pane of glass console to provision resources and seamlessly manage workloads running in a hybrid environment.

Sovereign DBaaS manageability aspects and differentiations span the following pillars:

Upgrades and patches

Users have complete control over the maintenance window and upgrade versions. Public cloud providers often have large windows of planned downtime to upgrade the underlying databases and the infrastructure. This downtime does not count towards the stated SLA but can interfere with organizations' workloads.

Cloud providers try to minimize disruption by picking off-peak hours in the regions where their DBaaS is deployed. But, with organizations becoming increasingly global, this is not an option.

✓ **Backup management**

Backups on public DBaaS are generally proprietary. For example, a backup of MySQL on Amazon RDS can't be restored on GCP Cloud SQL. The only way around it is to perform a slow, logical backup using MySQLdump.

Sovereign DBaaS uses native open-source DBaaS binaries and hence guarantees compatibility.

In addition, Sovereign DBaaS provides automatic backups and agreed-upon retention periods so that customers do not need to spend their resources on undifferentiated non-value-add tasks.

✓ **Monitoring and alerting**

Sovereign DBaaS integrates with the underlying cloud provider monitoring infrastructure, such as Amazon CloudWatch and Azure Monitor. The advantage they bring is that logs are in cloud vendor-neutral formats and can be consolidated in a single store to provide a merged view of operations.

✓ **High availability (HA) and disaster recovery (DR)**

Sovereign DBaaS permits high availability and disaster recovery by seamlessly deploying across any location. The open-source database replication engine has the advantage of being fully compatible across the deployment locations. This can prevent downtime in case of an unexpected failure of a cloud provider.

Database replication strategies are determined by the organization's business continuity process (BCP) requirements. They include defining the amount of data loss or the recovery point objective (RPO), and the amount of time needed to switch from the primary instance to the secondary one or the recovery time objective (RTO). Sovereign DBaaS instances use either synchronous or asynchronous replication depending on network latency, egress costs, and BCP requirements. The organization has full control over this management aspect.

✓ **Configuration management**

Every cloud provider enables only a select set of configuration parameters or extensions. In fact, most of them restrict root access. This inconsistency limits the ability to migrate from one cloud to another. Sovereign DBaaS enables the same configuration parameters regardless of where the DBaaS is deployed.

Sovereign DbaaS' single pane of glass management console further eases the management overhead by unifying and simplifying management activities.

Public DBaaS has been all the rage due to its ability to abstract the non-value-add management aspects from end-users. However, this tradeoff comes at the cost of losing control and price premium. Sovereign DBaaS shares management responsibilities with its customers. It provides all the fundamental management and automation building blocks with the addition of allowing the customers to turn the dials on how much automation they want.

Security

DBaaS uses the shared responsibility model that is common for the public cloud providers but with a difference. In public DBaaS, cloud providers take on a more significant role in ensuring security than Sovereign DBaaS providers. The latter is responsible for software security controls according to EAL 2 compliance and performing penetration tests, while the customers have a greater responsibility for ensuring data security.

The control plane is locked down. Providers ensure infrastructure security for data at rest and in motion, e.g., using VPN by default. However, customers bring their own keys (BYOK). As a result, the customers are responsible for all key management aspects, such as rotation and storage in audit vaults. Although key management (KMS) is the customers' responsibility, Sovereign DBaaS providers can maintain the keys in their KMS or HSM offerings.

In keeping with the emphasis on equipping customers with full control, Sovereign DBaaS allows customers to configure based on their own security needs and frameworks, such as complying with the customer's security model, e.g., ISO 27001.

Governance and privacy

Privacy is not only the final key characteristic of Sovereign DBaaS but one that gives it its name. It is the *raison d'être* of many organizations seeking higher control on where their entire end-to-end data stack is hosted. The reason why privacy is so vital is that compliance laws are in flux. For example, Schrems II invalidated private data transfer between the EU and the USA in July 2020. However, in March 2022, the EU and the US announced "The Data Privacy Framework" to strengthen privacy protections in

trans-Atlantic data flows, the details of which are expected to be finalized later in 2022. In addition, many countries are developing their own data residency laws, often modeled on the EU GDPR. These geographical compliances are in addition to the industry vertical regulations, such as HIPAA for healthcare and PCI-DSS for financial services.

Not only does Sovereign DBaaS provide users with full control over where their data resides, but they can easily migrate the data to a different location in the event a new compliance regulation demands it.

Sovereign DBaaS providers must have SOC 2 Level 2 certification to ensure that they adhere to the strict security guidelines. Providers offering services to other countries and institutions may have to comply with other regulations like FedRAMP certification for the US Government.

5

Benefits and challenges

Benefits

This research paper has listed many of the benefits of Sovereign DBaaS. In summary, it takes the best of public DBaaS, such as elasticity and automation, without the tradeoff of being locked into a single vendor's solution. Some of the benefits are as follows:

✓ **Cost Saving**

A key benefit of Sovereign DBaaS is to rein in the increasing cost of public cloud deployments. Typically, Sovereign DBaaS cost structure is less than that of the hyperscalers. The savings become more apparent as the scale goes up.

✓ **Control**

Sovereign DBaaS is mainly about ownership and control over operational tasks. It can be deployed on private cloud on-premises and in public cloud—in customers' own VPC or in the vendor's VPC to meet organizations' specific compliance and cost requirements.

✓ **Flexibility**

Sovereign DBaaS provides extreme flexibility in configuring and tuning databases.

✓ **Latency**

Sovereign DBaaS can meet the strict quality of service needs for latency and throughput performance and guarantee predictability.

✓ **Manageability**

A single pane of glass is used to manage various deployments. The overhead, like installation, provisioning, upgrades, patching, backups, etc., do not add value to the organization's strategic imperatives and can be handled automatically by the Sovereign DBaaS provider.

The initial beneficiaries of Sovereign DBaaS were organizations in the regulated industries such as financial services, insurance, healthcare, and government. However, as the offerings have matured, other industries are ramping up their use of Sovereign DBaaS.

Challenges

Public DBaaS capabilities from hyperscalers are much better understood compared to the emerging and nascent Sovereign DBaaS category. Some of the challenges of Sovereign DBaaS are related to the lack of understanding of its maturity, which include:

✔ **Vendor viability.**

As Sovereign DBaaS is a new category, buyers are concerned about the viability of vendors, their revenue, years in business, number of customers, and funding status.

✔ **Customers.**

Ratings of Sovereign DBaaS products on various public rating websites, such as Gartner's Peer Insights, G2, and TrustRadius, are not as readily available as public DBaaS.

✔ **Identification of cost savings.**

Public cloud DBaaS pricing is well known, but identifying cost parameters is less well known. Its cost includes human as well as technical.

✔ **Support.**

Support, especially for globally distributed deployments and in multiple languages, may be limited due to the lack of maturity of the category. Sovereign DBaaS vendors should provide multiple levels of support, including rapid prototyping of the end-to-end solution.



Sovereign DBaaS vendors

Sovereign DBaaS vendors

Sovereign DBaaS vendors sell their products in multiple ways; direct to the end customers or indirectly through OEMs like Cisco or partner agreements with CSPs. Often the CSPs may be the lower-priced options, like Linode (part of Akamai), Vultr, or Digital Ocean. Depending upon the vendors' go-to-market deployments, cost models may vary between subscription and pay-as-you-go utility consumption.

Several vendors have some of the capabilities of Sovereign DBaaS but don't meet all of its requirements:

- ✓ **Hyperscale public cloud providers**, such as AWS Outposts, provide hybrid extensions to their public cloud offerings to meet data residency requirements. However, hyperscalers control the hardware infrastructure leading to vendor lock-in.
- ✓ **Managed service providers**, such as Instaclustr (acquired by NetApp in 2022).
- ✓ **Specialized vendors** include Scalegrid, Nutanix, and Equinix.
- ✓ **Do-it-yourself (DIY)** or in-house solutions built by organizations. However, it requires skills and can incur technical debt to keep them updated with the latest technology advancements.

Severalnines has been at the forefront of open-source database operations automation development for over a decade and is a provider of Sovereign DBaaS through its ClusterControl and CCX solutions.

Conclusion

Sovereign DBaaS' time has come. It has the potential to meet strict data privacy compliance guidelines. The main reason one would consider this option is to assert more control over their data stack, which helps meet compliance guidelines, reduce business risks and avoid runaway cloud costs. It leverages key benefits of public DBaaS but without the tradeoffs on the control of its location and ownership. Figure 3 shows a SWOT analysis of Sovereign DBaaS.



Figure 3. Sovereign DbaaS has compelling strengths that appeal to enterprises that are regulated and have large amounts of sensitive data. Its biggest weakness is that being a relatively new category, there is not enough awareness of its capabilities.

The cornerstones of Sovereign DBaaS are open-source databases and a consolidated management console. It helps realize the promise of a truly hybrid multi-cloud environment without vendor or environment lock-in. End users not only choose which environment they want to run their workloads, but they have the freedom to migrate to the one that meets their security, cost, and performance needs.

There is already a growing deployment of Sovereign DBaaS from enterprises that have a footprint in multiple geographical regions, must adhere to strict regulations, and are cost-conscious.

**Want to see how
you can create
your own
Sovereign
DBaaS?**

**Schedule a
consultation.**