

Welcoming Portworx to Pure Storage



Delivering Kubernetes Data Services for Tomorrow's Cloud Native Applications

We're excited to announce Pure Storage's intent to acquire Portworx. Portworx brings with it a talented team of engineers and technologists. Its customers are some of the most innovative organizations building cloud-native applications in the market today.

The Portworx team, led by CEO Murli Thirumale and co-founders Goutham Rao and Vinod Jayaraman, will form a new Cloud Native Business Unit inside Pure focused on driving the premier data-services experience for Kubernetes-based applications. Let me give you a window into why we're so excited to bring Portworx into Pure, what it means for our collective customers, and where we're going from here.

Every Major Architecture Shift Creates a New Leader

It has become all-too-obvious that a major new vendor arises with every major IT architecture transition—client-server, web, virtualization, cloud, etc. These are the vendors that understood the fundamental shift and *purpose-built* a new product for the new world, instead of just trying to retrofit what they had.

Pure was born in one of these shifts. In our case, it was the shift from spinning disk to all-flash storage. We

pioneered a model of the all-flash array that became the blueprint for the industry to follow (or at least try to). We succeeded by embracing the “10x” opportunity that a major shift creates, tried to clear all legacy thinking and approaches from our whiteboards, and designed for the new world.

Kubernetes + Containers: The Multidimensional Shift

Containers are fundamentally re-wiring how people think about designing infrastructure. But that’s really only one dimension of the architecture transformation. Containers aren’t just changing how infrastructure design, they’re changing how we write and package applications, fundamental to the application architecture itself. They’re also changing how we design for the cloud, making it possible to build applications to be hybrid and multicloud by design from the beginning.

Add onto this a parallel innovation—Kubernetes—which is rapidly becoming the de facto orchestration engine for modern cloud native applications. In fact, we believe Kubernetes will expand beyond containers to be the orchestration engine for the full stack of tomorrow’s application infrastructure.

So, if there’s a revolution on three fronts—application architecture, infrastructure design, and multicloud architecture—it seems a bit silly to think that today’s storage and data-management solutions will be capable of evolving to match the true power of containers and Kubernetes. Never has there been a better opportunity for a *purpose-built* solution.

Lessons Learned in Container (Storage) Evolution

We saw these trends early at Pure. We began investing years ago to build industry-leading platform support for containers and Kubernetes, first by investing in a FlexVolume driver, then through the CSI spec, and ultimately by reaching beyond CSI with [Pure Service Orchestrator™](#).

Pure Service Orchestrator virtualizes many FlashArray™, FlashBlade®, and Pure Cloud Block Store™ instances into a single storage pool. It enables developers and Kubernetes-managed applications to seamlessly request persistent volumes and interact with storage services in a completely Kubernetes-native way—without any knowledge of the underlying infrastructure (just as it should be). Pure Service Orchestrator has helped make containers and cloud-native apps one of the fastest-growing use cases on the Pure platform. But as we dug deeper into these use cases in active use, we found that appliance-based storage wasn’t a perfect fit for many of the ways that customers wanted to use containers.

- Most Kubernetes projects start small. A developer is asked to start exploring Kubernetes and containers and they do so with whatever infrastructure is available: in a lab or on some cloud instances. And before you know it, you have more containers—and complexity—than you ever imagined/predicted.
- Many cloud-native applications are born and built in the cloud, and may never come on-prem. If a migration is going to happen, it’s between cloud Kubernetes services to become multicloud.
- The Kubernetes religion is software-centric. For many, the first instinct is to look for a software solution that runs on bare metal infrastructure. This not only supports the agility that containers require but makes it easier to have a consistent architecture between on-prem bare metal and cloud compute/storage instances.

In all these situations, hardware-based storage solutions are either impossible to use, inconvenient to use, or just don’t align with the goals of the Kubernetes project. You want something more software-centric, cloud agile, and *purpose-built* for Kubernetes... or you may just not want to wait for IT to respond with

infrastructure when you have to move fast.

Along this journey, we started to notice something interesting. Several of the largest Pure container customers were running a software-defined storage layer from a company called Portworx on top of Pure arrays. Companies like T-Mobile and Charter Communications were doing this at very large scale, using this architecture to deliver persistent storage to their mission-critical stateful applications. Something interesting is going on here indeed.

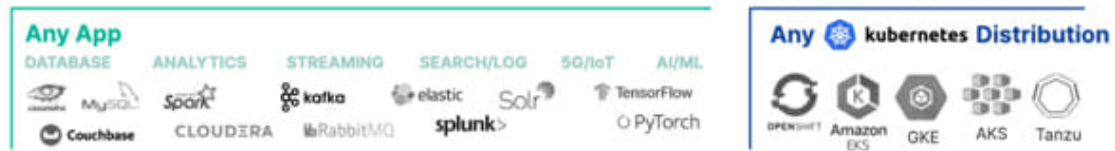
Enter Portworx: The Purpose-built Kubernetes Data Services Platform

I won't attempt to tell the Portworx team's founding story, they do it better in their companion posts. Suffice it to say that Portworx was founded to take advantage of the container transition and create the perfect *purpose-built* data services platform for Kubernetes. Several key design decisions that permeate the product.

- **100% software:** This enables Portworx to deliver container data services in a consistent manner, no matter what infrastructure you choose. It also enables Portworx to scale up to the largest enterprise container deployments, scale down to the smallest 5G edge site, and support everything in between.
- **Simple, declarative, and Kubernetes-native:** Cloud native developers simply don't want to have to understand infrastructure. The perfect infrastructure is one that is invisible, and "just works" as an extension of Kubernetes. Define declarative storage classes and go. This was the design principle of Portworx and it's what makes Kubernetes admins and developers *love* Portworx.
- **Infrastructure-agnostic:** The whole point of Kubernetes is to package applications to be portable, so it's key that Portworx is similarly infrastructure-agnostic. This means any cloud, any Kubernetes distribution, and any underlying storage infrastructure: in cloud or on-prem (or ideally both), virtualized, bare metal, or as-a-service.
- **Enterprise-class:** This one is critical. As Kubernetes matures and customers evolve from stateless to stateful applications, Kubernetes and cloud native applications need to be as reliable as the traditional application infrastructure that predates them. You can't step backward in reliability to move forward in architecture. This is a **real** struggle today. Enterprises want all the capabilities that they've been used to in traditional platforms:
 - Failure protection against drives, nodes, racks, and sites
 - Backup and recovery of complete applications
 - Synchronous and asynchronous disaster recovery between AZs and regions
 - Rich role-based access controls (RBAC) for multi-tenancy clusters
 - Consistent encryption and security that goes wherever your application does

Portworx was designed to bring enterprise-class resiliency to the Kubernetes world—but in a way that didn't destroy its benefits along the way.

The result of these efforts—now six years in—is the most mature, capable, and complete platform for delivering Kubernetes data services.




The industry's leading Kubernetes Data Services Platform
 for building, automating, protecting, and securing Cloud Native applications.



Not Just an SDS Layer—a Data Services Platform

There are several options for software-defined storage under Kubernetes. The most obvious is probably the open source tool CEPH, now also packaged as RedHat OpenShift OCS. The problem is that containerized applications need more than just software-defined persistent volumes. Persistent Kubernetes applications need a whole suite of enterprise-class data services delivered with Kubernetes agility. That's exactly what Portworx delivers.

- **PX-Store:** The software-defined storage layer can turn any commodity storage (SSDs, HDDs, and cloud storage) or enterprise arrays (Pure FlashArray, NetApp, EMC, etc) into reliable data services under complete control of Kubernetes. PX-Store gives everyone—developers, applications, and infrastructure teams—a consistent, policy/service-class/API view into data services.
- **PX-Backup:** Kubernetes applications are distributed and complex. Protecting a Kubernetes-based application means not only backing up the application itself, but also all the cluster artifacts, configuration, controllers, and other metadata. PX-Backup is designed to work with PX-Store, but doesn't require it, and can backup any native-cloud storage on AWS, Google, or Azure.
- **PX-Migrate:** Cluster migration is a daily occurrence for Kubernetes admins whether an application is being moved between clouds, deployed into the prod cluster, or just because of cluster maintenance. PX-Migrate simplifies such migrations, enabling seamless movement of applications between heterogeneous Kubernetes clusters.
- **PX-Secure:** As enterprises adopt hybrid-cloud and shared Kubernetes clusters, PX-Secure

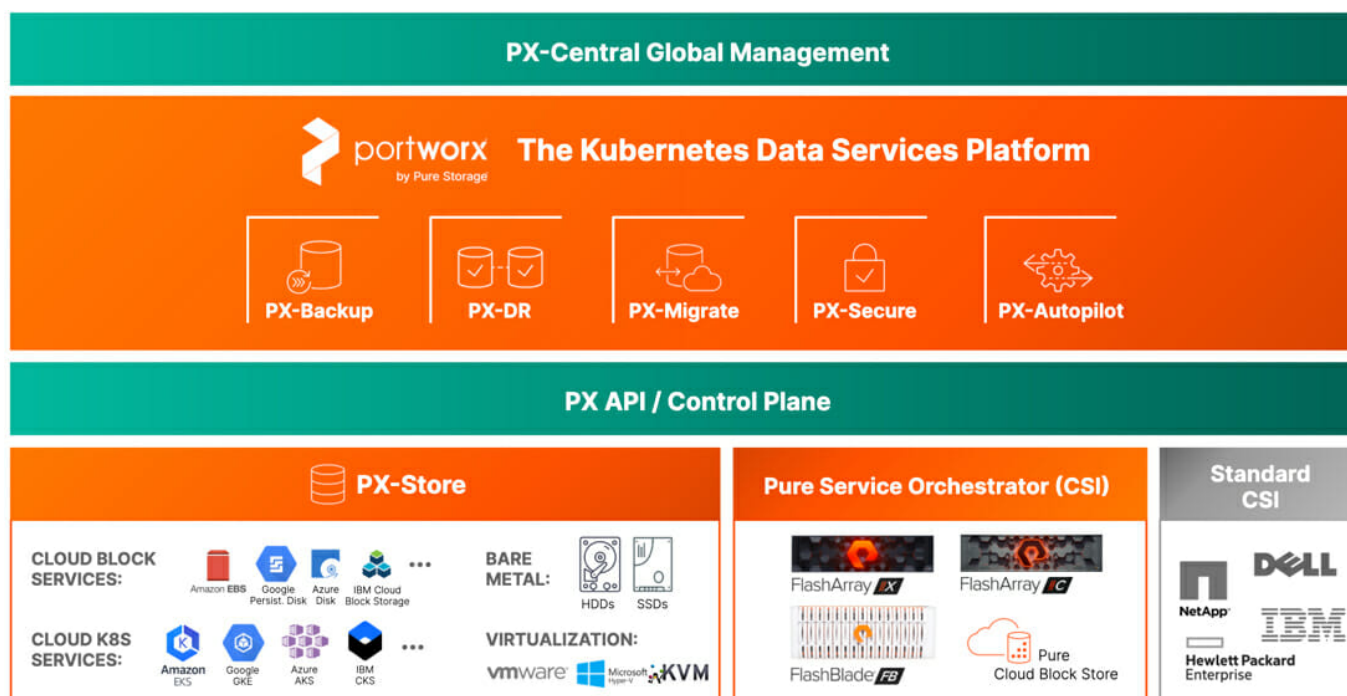
delivers robust, consistent encryption and rich RBACs, simplifying the ability to exert compliance and security controls on Kubernetes-based applications.

- **PX-Autopilot:** As infrastructure itself—especially in the cloud—becomes more agile, intelligent auto-scaling has the potential to deliver dramatic efficiency gains. PX-Autopilot watches your clusters and makes smart decisions when necessary to keep your clusters available: auto-growing storage resources on a just-in-time basis, for example.

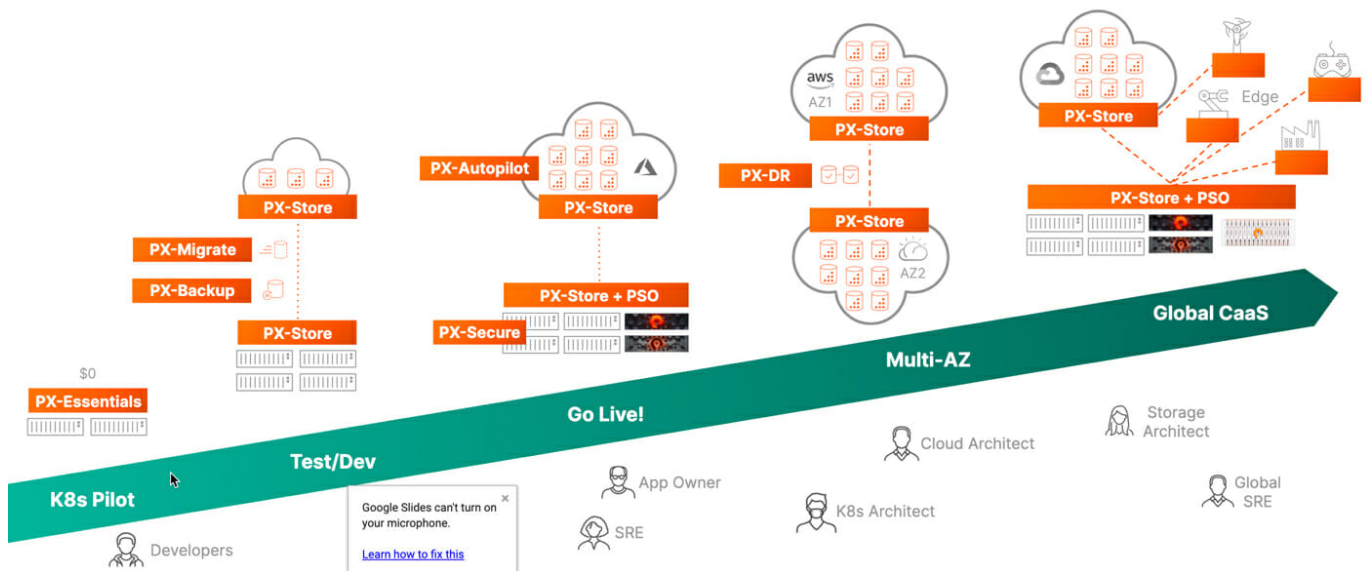
Portworx + Pure: How 1 + 1 = 3

Two hallmarks of the Kubernetes movements are openness and choice. By bringing together Portworx and Pure, we can offer you the most comprehensive set of storage options for running your Kubernetes clusters—all orchestrated from a single PX control plane.

Our plan is to deeply integrate Portworx with Pure Service Orchestrator to create a seamless combination of Pure and Portworx. (Stay tuned for more information once we close the acquisition.)



The path to Kubernetes and cloud-native applications is a journey. We're confident that no single storage solution can be the right fit for every stage along the way. In fact, if you're going big with containers, it's likely that different projects and groups are at various phases of the cloud-native journey. Pure and Portworx together can deliver the perfect data service for every stage.



What can you do now? You can now set up your very first Kubernetes cluster and power it with PX-Essentials—[free to download](#) and free forever—for up to five nodes and 5TB of data. As you look to develop across clouds in Kubernetes, the PX-Store, PX-Migrate, and PX-Backup elements come into play to easily version and distribute dev instances. As you go live, you can scale the Portworx environment scaled both on-prem and in the cloud—and you may consider deploying on a Pure all-flash array for higher performance and resiliency. PX-DR then enables multi-AZ or multi-region deployments for enterprise-grade resiliency.

As you scale containers to a global container-as-a-service offering, the entire suite works together to deliver the right data services for each site and use case. Pure’s job is to ensure that it’s a seamless, Kubernetes-native experience—whatever storage you choose.

What We Will—and More Importantly *Won’t*—Do

We will:

- Run Portworx as a separate product line within Pure with a heavy dose of autonomy
- Increase investment in Portworx engineering to accelerate its roadmap
- Continue to support heterogeneous infrastructure at every layer—including running on any cloud, bare metal, or storage-array footprint (even Pure’s competitors!)
- Introduce Portworx into Pure’s global sales channel, helping expand the Portworx global footprint
- Leverage Pure’s global-support organization to scale Portworx support
- Continue investment in the open-source community around Kubernetes, making frequent code and architecture contributions as both Pure and Portworx have done with CSI, stork, and libopenstorage
- Integrate Portworx with Pure Service Orchestrator to build the industry’s most complete Kubernetes data-services platform
- Keep both companies **orange**—in color and culture!

We won’t:

- Force Portworx customers to deploy on Pure hardware solutions. Open means open!
- Sacrifice Portworx’s roadmap to invest in Pure integration. We plan to integrate where prudent, but support the Portworx plan to continue building out the data-services platform. (And we want

that to go faster.)

- Destroy Portworx support. Portworx customers love Portworx support and Pure is renowned for world-class support as well. We'll make this a great joint experience.

Now Available in a Kubernetes Cluster Near You

I hope you get a sense for what an exciting day this is for all of us at Pure and Portworx.

The best part: While we're sure to do deeper integration in the future, there's no need to wait to get going with Kubernetes and containers on Pure today. The Portworx [Kubernetes](#) Data Services Platform is generally available and ready for you to use on any Kubernetes cluster. Both Pure FlashArray and FlashBlade fully support containers and you automate them in your Kubernetes clusters via Pure ServiceOrchestrator or Portworx today. And you can get going by [signing up for PX-Essentials](#) today!

More About Today's News from Our Execs

["Pure to Acquire Portworx, Powering the Future of Cloud Native Apps"](#)

Charlie Giancarlo, Pure Storage Chairman and CEO

["Get Ready for An Even More Exciting Future - Portworx is Being Acquired!"](#)

Murli Thirumale, Portworx Co-founder and CEO

["Apps drive infrastructure. That's why Portworx is joining Pure Storage."](#)

Goutham Rao, Portworx Co-founder and CTO

["The power of a truly cloud-native storage infrastructure"](#)

Vinod Jayaraman, Portworx Co-founder and Chief Architect

Additional Resources

[Pure Storage Press Release](#)

[Pure Storage + Portworx FAQ](#)

[Portworx Blog](#)