PUBLIC CLOUD AS COMPONENTS FOR PURE CLOUD BLOCK STORE™

pivotnine

PURESTORAGE®

PURE STORAGE IS A PIVOTNINE CLIENT

AUTHOR: JUSTIN WARREN

JULY 2021

01 PUBLIC CLOUD AS INFRASTRUCTURE COMPONENTS

Pure Storage has used the cloud as infrastructure building blocks for Pure Cloud Block Store, providing a way to abstract the cloud and take advantage of cloud-specific advantages, while also providing benefits of software from Pure that are greater than what the raw cloud provides.

In this whitepaper we investigate how Pure Storage has constructed Pure Cloud Block Store and why it makes sense. We look at the benefits of Pure's approach, and the potential downsides or risks.

Every technology has strengths and weaknesses and we are here to explain what Pure Cloud Block Store is best at and the kinds of workloads it makes a good fit for.

Pure Cloud Block Store on AWS

Pure Storage first constructed Pure Cloud Block Store on AWS, using AWS services as primitives in much the same way that Pure uses hardware components as primitives for its storage arrays.

For active storage, Pure Cloud Block Store uses AWS i3en instances as virtual drives and combines a set of seven virtual drives into a virtual shelf. The first seven virtual drives are also connected to an Amazon EBS io2 volume as NVRAM, which is shared across the Pure Cloud Block Store instance; additional virtual shelves do not require additional io2 instances attached. This virtual shelf is where active data is stored for most of the operational work that Pure Cloud Block Store does.

The i3en instance type was chosen mostly for its high-speed network access (from 25Gbps and up to 100 Gbps at time of writing) while the io2 instance is optimised for storage IOPS.

Data on the virtual drives is kept protected and highly available with RAID to guard against up to two drives failing at once, and failed drives are rebuilt automatically. Pure also uses a spread placement group to ensure virtual drives are in separate failure domains (racks) within AWS' datacenter. This is mostly to ensure that data remains online and available as expected operational failures occur.

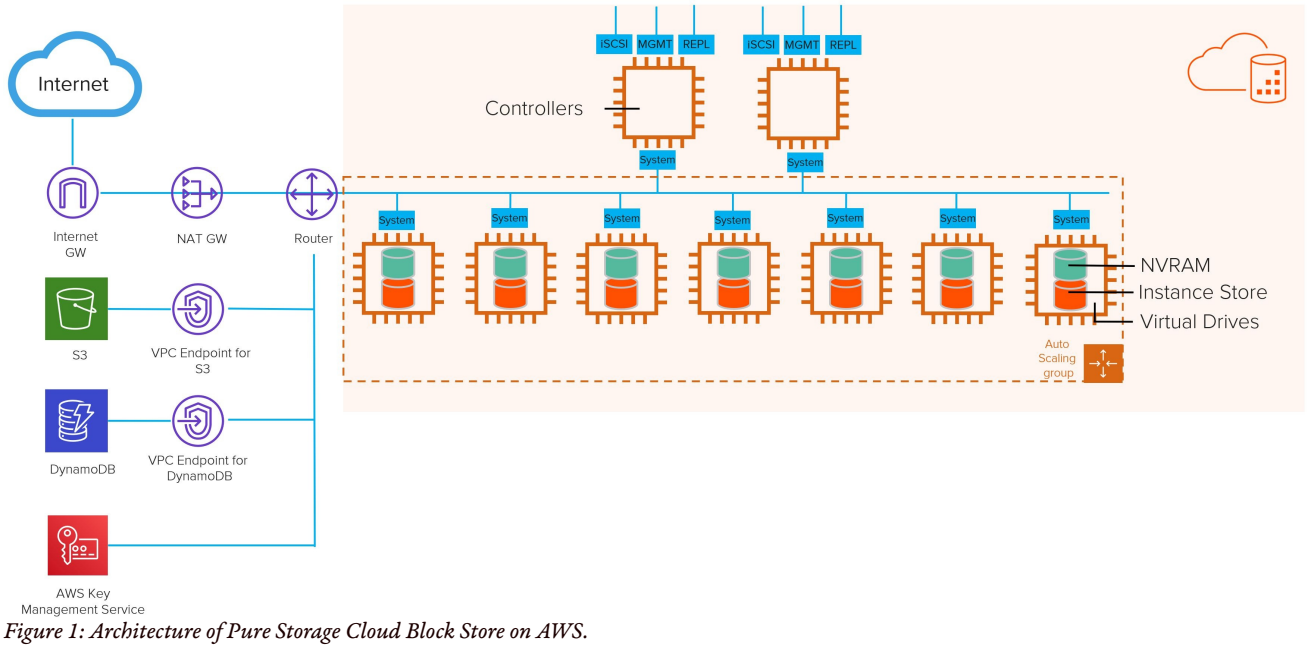


Figure 1: Architecture of Pure Storage Cloud Block Store on AWS.

01

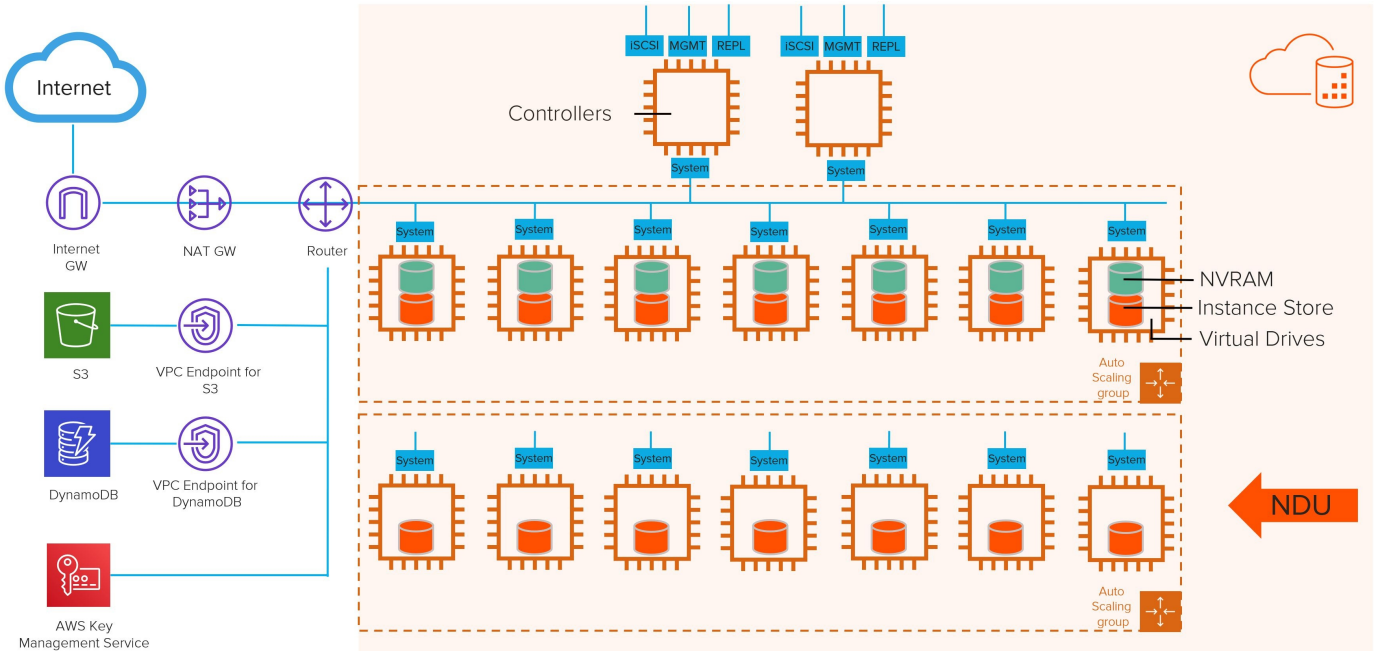


Figure 2: Architecture of Pure Storage Cloud Block Store on AWS during non-disruptive upgrade.

To further protect the active data, it is persisted to Amazon S3 which plays the role of the flash media of a physical Pure Storage array. This is where data is written to after Pure's operating system magic massages the data to deduplicate it, compress it, and generally optimise it for longer-term storage. Pure Cloud Block Store could lose all of the controller and virtual shelf layers without losing data, just as losing a controller or NVRAM on a physical array would not affect any data on the flash.

The architecture of Pure Cloud Block Store is deliberately composed of pieces of cloud infrastructure in order to take advantage of what makes Pure Storage's Purity operating system special, while also benefitting from what makes the components themselves special.

This approach helped tremendously when Pure wanted to provide Pure Cloud Block Store to customers on Microsoft Azure.

01

Pure Cloud Block Store on Azure

When building Pure Cloud Block Store for Azure, Pure revisited the overall design and tweaked it to suit the differences in Azure compared to AWS.

In Azure, Pure again chose VMs for controller nodes, and a dual-controller architecture, mirroring the architecture of physical arrays. But instead of having virtual drives external to the controllers, Pure made use of Azure's Ultra SSDs as both NVRAM and persistence layer. Two Ultra SSDs make up the NVRAM layer, and 14 Ultra SSDs provide the persistent storage layer, and each Ultra SSD is dual-connected to both controller VMs.

When Pure built Pure Cloud Block Store for AWS, AWS did not have the ability to multi-attach storage to the controllers, but when it came time to build for Azure, this functionality was available, so Pure used it. AWS now has some support for multi-attach for some types of storage, but it's limited, and it doesn't have the kind of reservation support that Pure would need to implement the same architecture in AWS as used here for Azure.

Pure is looking at making changes as AWS invests in its technologies, and may end up making use of Azure Blob Store in a similar way to S3, though it has no particular plans to do so. The key point here is that customers don't really need to care about these kinds of decisions because of the architecture of Pure Cloud Block Store.

Just as in Pure Cloud Block Store for AWS, Pure Cloud Block Store for Azure looks like any other Pure product. It runs Purity, supports the same APIs, talks to Pure1 and integrates with all the other tools and techniques that customers have invested in for managing their physical arrays.

Pure Cloud Block Store is, like most infrastructure software, an abstraction of physical hardware.

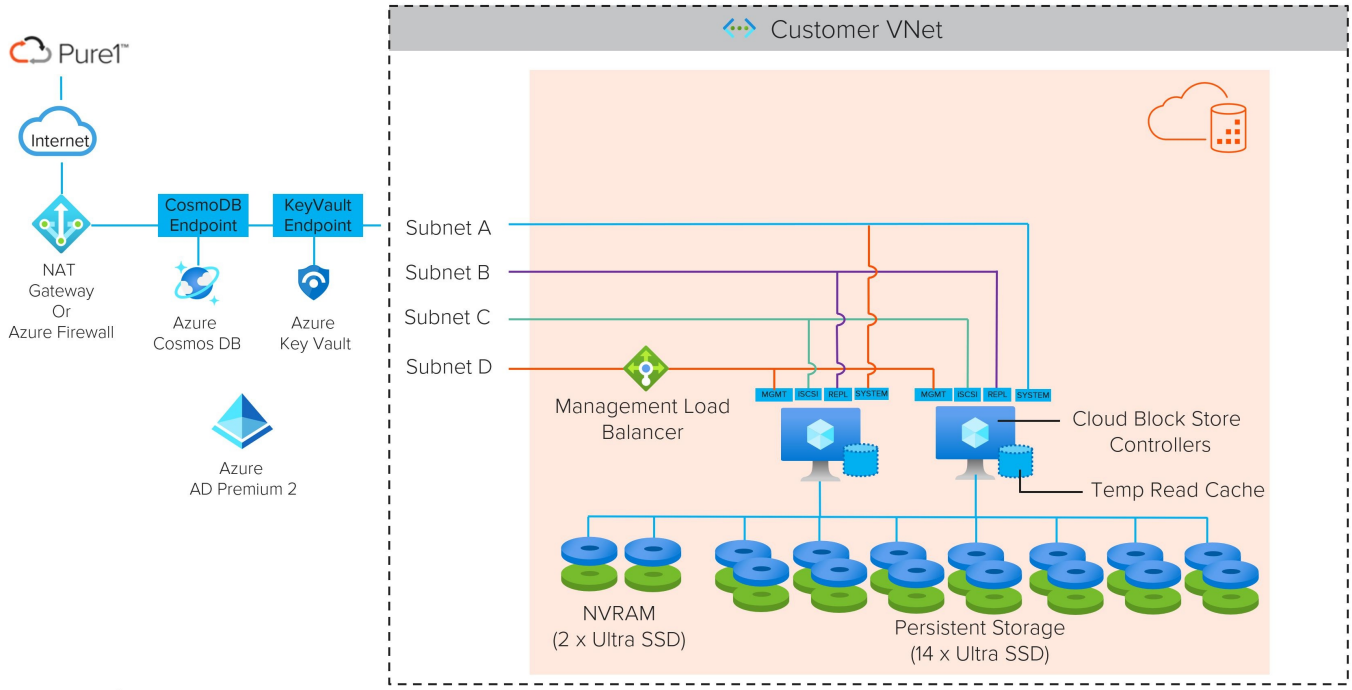


Figure 3: Pure Storage Cloud Block Store Architecture on Azure.

CLOUD AS COMPONENTS

02

While we don't claim that treating cloud services as components on which you build a service like Pure Cloud Block Store is a particularly novel insight, PivotNine believes it is not given as much attention and weight by the industry generally as it deserves.

There is nothing inherently better about a service constructed in this way compared to software written for virtual machines running on, say, VMware or Virtualbox that happens to be running on a VM instance in a public cloud. Unless there is infrastructure specific code, it likely makes no difference whatsoever, and claims of 'cloud-native' being inherently superior are overblown.

However, if a vendor can demonstrate concretely that its architecture takes advantage of cloud components in a specific way, the case is stronger.

We believe that the most important benefit is not specific to the speeds and feeds of software tuned for the specific infrastructure environment, but that the architecture of the software allows for this tuning to occur transparently to the customer. It is a form of non-disruptive upgrade if the customer can rely on Pure Cloud Block Store functioning the same way in AWS or Azure with at least a minimum expected performance level, and one that will continuously improve as the underlying components improve.

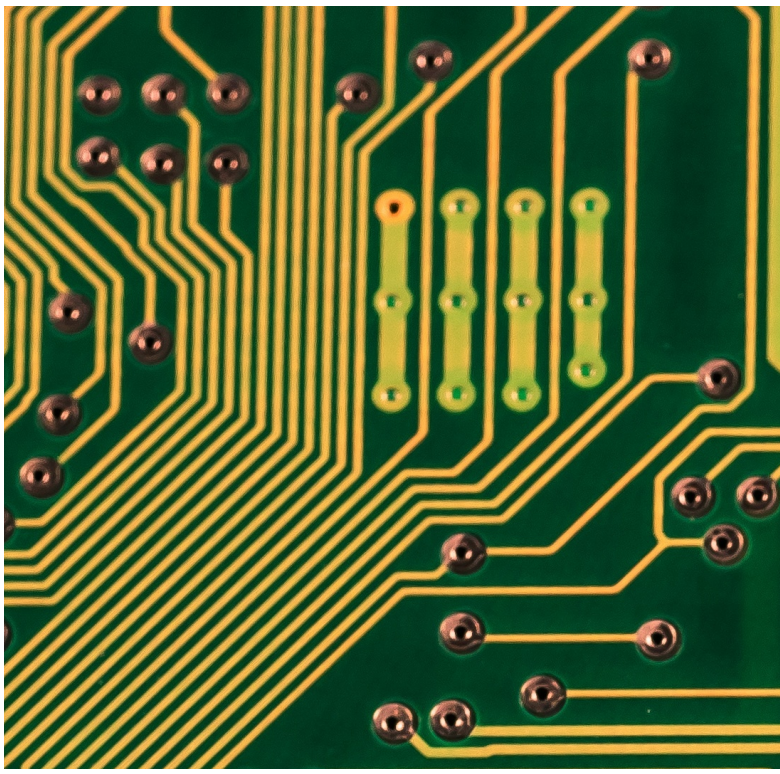
These are the benefits we see from physical infrastructure upgrades. The greater flexibility of cloud-based software-defined infrastructure means these kinds of seamless upgrades in capability should be expected, not merely appreciated.

This kind of continuous improvement should be a compelling reason to choose a vendor provided service, rather than building a service yourself from the raw components provided by a cloud. It is, in fact, exactly the same reason underpinning the choice of a cloud vendor's components, just at a higher level of abstraction.

The Hardware-Software Continuum

A hard distinction between hardware and software is a tricky thing, particularly today. Any characteristic we might assign to hardware, such as being immutable or relative difficult to change, can often be applied just as effectively to software, depending on the circumstances. Our working definition is that hardware is physical components that requires a physical intervention to make changes at one extreme end of "difficult to change" continuum, while remotely accessible changes to the internal state of a computer system that are extremely easy to make is at the other extreme.

We can place different kinds of systems somewhere on this continuum, and their absolute position is less interesting than their location relative to one another.

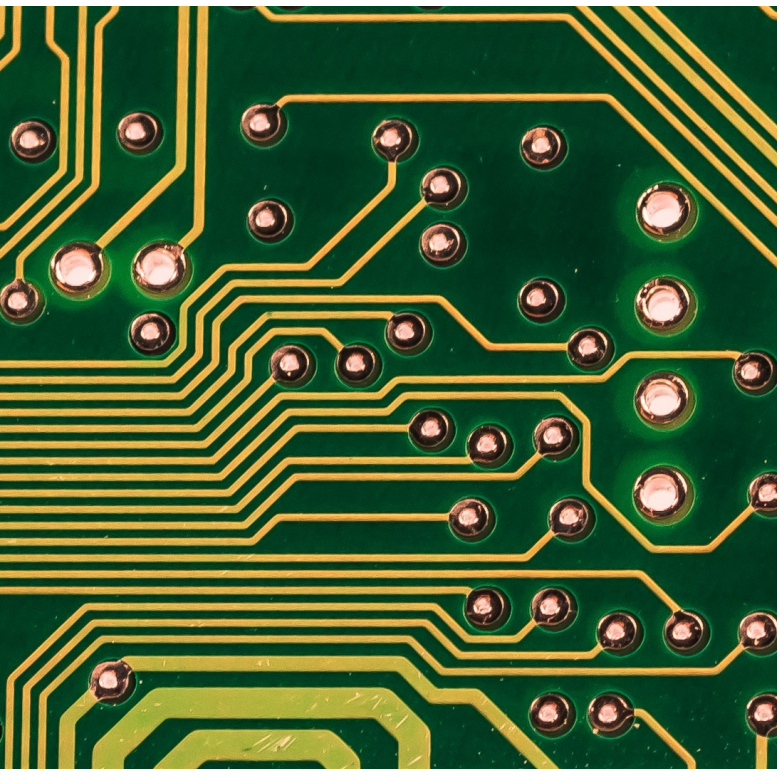


02

An ASIC and an FPGA may perform the same calculations, but we make different tradeoffs when deciding which to use. An ASIC takes a long time to design and make, and once manufactured cannot be changed, but runs much faster than an FPGA. An FPGA, meanwhile can be changed, so single-task speed is sacrificed for greater flexibility.

Hardware itself is rarely purely hardware these days, given the prevalence of embedded operating systems and dynamic hardware like FPGAs. Indeed, Pure has made use of FPGAs over ASICs in some sub-assemblies of its storage arrays to provide for flexibility, such as being able to switch between different NAND flash components from different manufacturers that update at different times through the year.

The decisions made by Pure insulates the customer from deciding which manufacturer of NAND flash to use to provide the raw capacity of the array. Indeed, if Pure does its job well, the customer will never need to know or care. The entire point of engaging a vendor to do a job is so that you don't need to do the job yourself. Personally, I don't want to have to manage the supply chain relationships involved in modern NAND flash manufacturing in order to buy an SSD for my laptop. I would get nothing else done.



Cloud already provides a lot of this kind of functionality. Indeed, it is one of its great benefits. One needn't worry about purchasing, shipping, or installation of a server in order to use one for a few hours, or all month.

For most of the work we want to do with computers, we don't really need to care very much about which manufacturer AWS or Azure use for their servers, networks, or storage. Ideally, we don't need to care at all and just rent "some computer" to run my software.

Pure has embraced this aspect of cloud to provide value to its customers with Pure Cloud Block Store and, we argue, it's the kind of behaviour we should expect to see from vendors.

How To Add Value

Pure Cloud Block Store is an example of truly embracing the cloud. It is not a mere lift-and-shift version of software running as a VM that happens to be in a public cloud but could just as easily be in an on-site VMware environment. It is not just a thin virtualization layer on top of already abstracted services.

Pure Cloud Block Store treats cloud resources like infrastructure just as physical servers treat microprocessors and RAM and flash and clock circuits as infrastructure, and how datacenters treat power and cooling and network interconnects as infrastructure. They are building blocks for building something else not and end in and of themselves. The point of AWS isn't to

02

be AWS, it is to provide customers with building blocks for constructing other things.

And while you can construct a simple, clumsy copy of what you already know: a server running under a desk, or perhaps even rack-mounted in a datacenter, something to run software exactly the same way you always have, that doesn't really take advantage of what cloud is about. There isn't zero value in doing this, but the power of cloud comes from embracing what makes it different.

There are plenty of options for software defined storage (SDS) that you can decide to run in a cloud VM instead of a VM in an on-site VMware cluster. They work, but in order to do so, they necessarily limit themselves to the features and functions that they know will be available everywhere. They choose compatibility over optimization, and this makes sense for what they're trying to do.

The compatibility with existing processes may well provide so much value that it dwarfs the minor inconvenience of a little extra latency, or forgoing certain features. Worrying about the details of speeds and feeds shouldn't come first, and may not actually matter at all, in the final analysis.

What should be front of mind is what you're actually trying to accomplish. What outcome are you trying to achieve, and what is important to consider when achieving that outcome? What tradeoffs will be needed to be made, and where should compromises be struck? What can you live with, and what can't you live without?

Perhaps a few extra milliseconds of latency is worth a saving of $10k a month? Or maybe you're happy to spend $20k a month to run twice as many jobs in a test run?

03

WHY USE CLOUD BLOCK STORE?

Pure Cloud Block Store provides value in three key areas, in our view:

1. Pure Cloud Block Store is optimized for cloud workloads.
2. Pure Cloud Block Store continuously improves.
3. The Pure Cloud Block Store interface is consistent with other Pure Storage products.

Optimized For Cloud

The first area is that Pure Cloud Block Store is optimized for cloud.

There are workloads that benefit from being in the cloud, such as test and dev, where the ephemeral, elastic nature of the cloud is a tremendous boon. Yet the clouds themselves do not provide the full functionality of an enterprise storage array.

For some workloads, such as parallel execution on replicas of large datasets, the way Pure Cloud Block Store is designed represents a cost saving compared to raw cloud services due to data deduplication, thin provisioning, and judicious usage of snapshots or clones. Pure Storage claims simple cash savings of 20-30% without needing any complicated TCO model shenanigans to compare apples with oranges, and with equivalent or better performance.

Building a similar service yourself, while possible, is not in most organization's best interests. Vendors exist to provide services that we cannot readily do ourselves in a cost-effective fashion. Our time is better spent focusing on whatever it is our customers want from us, and that tends not to be implementing storage deduplication algorithms.

Continuous Improvement

The second area is that Pure Cloud Block Store provides an abstraction of the cloud in the same way that the cloud is an abstraction of physical infrastructure. This abstraction allows the underlying components to change without you, the customer, having to care.

A physical array is an abstraction of the components of the array, such as CPUs and RAM and physical flash storage. The job of choosing which ASIC or FPGA or CPU to use, which RAM vendor should be used for this array, and which flash foundry to source the chips from, are all left to Pure. As customers, we are not involved in this decision process, and we don't need to be. That is Pure's role as a vendor.

If Pure Storage buys its flash chips from a different foundry, or changes from over-provisioned TLC flash to QLC flash (which happened in the C-series) you don't need to care, provided the interface you use—the storage array—behaves in the same way and the service level meets expectations.

In fact, you reap the benefits of improvements in underlying technology. When Pure moved to using cheaper QLC flash instead of TLC for capacity storage, it was able to pass on savings to customers without breaking its margins or degrading the performance of the array. Doing this well is a complex undertaking, and not the sort of this most customers should want to undertake for themselves.

Pure Cloud Block Store does exactly the same thing, but for cloud infrastructure. We can stop caring about which of the zillion instance types AWS has decided to announce this week, and which would be suitable for our workload, and just use some Pure Cloud Block Store. We can trust that Pure Cloud Block Store in Azure will be just as good, and get on with our day instead of spending it buried in hardware compatibility lists. We can spend our time using infrastructure to do things rather than solving infrastructure problems.

03

Interface Consistency

The third area is that it looks the same everywhere. This provides a consistent interface to storage services no matter where they are deployed. Management at scale requires this level of consistency because it reduces operational risk.

Reducing risk in this way means you can increase risk in other, important areas where the payoff is likely to be worth the increased risk. This is the same logic behind mostly using the same model of server, except for one or two special projects with special needs. Or mostly using one cloud for everything, except for a couple of systems that need to be on site, and for good reasons.

> *All stable systems resist change because that's what makes a system stable.*

Removing all of the downside risk you can frees up time, resources, and mental capacity to maximize upside risk in one or two places. Successful entrepreneurs understand this, almost instinctively. It doesn't guarantee success, but it vastly reduces the odds of failure.

But the cloud doesn't do everything, nor should it be expected to. Some workloads need to live on-site, and by adopting the same Purity interface for the storage regardless of where it sits, you can keep your operational risk low no matter where the workloads need to live. You can have, at least in a lot of places, the same consistent approach to management and operations.

You can also change your mind.

Interface consistency means that if you decide to move a workload from one place to another, you can rely on the same methods and processes you use elsewhere. You don't have to adopt a new method if it doesn't make sense to do so. Pure's replication techniques work the same on physical arrays as with Pure Cloud Block Store, which eases the challenging task of moving data between clouds or from on-site to cloud. It makes it easier to try something in the cloud before moving it back on-site, or to decide that you're now ready to move a workload from on-site to cloud, but keep all your carefully configured operational processes the same. You don't have to use this method, but isn't it nice to have the option?

Enterprise Process

A point often overlooked by cloud-native people, particularly those who have not spent much time in long-established enterprises, is that reliable processes matter at scale. Having different processes for different things creates confusing variance and opportunities for things to go wrong.

All stable systems resist change because that's what makes a system stable, and enterprises have amassed a substantial amount of stability. Changing them is challenging, and takes time, if you want to move from one stable system to another stable system without destroying the enterprise.

03

Yet complete stasis is also bad. Some change is necessary because the world changes. Competitors come out with new products, people decide they no longer like flared trousers and now want sugar-free drinks.

In enterprises, a substantial challenge of management is determining the level of dynamic tension between moving and staying still; constantly adjusting the amount of allowable change to adapt to changing circumstances so that it is never too little (leading to stasis) or too much (which tear the system apart).

An option that provides for similarity of process while also providing flexibility is an attractive option. Pure Cloud Block Store provides something that behaves similarly to Pure Storage appliances wherever they may be, but is also 'cloud-like' in its flexibility.

Cloud Lags on Features

For all the hype of cloud, it tends to lag on features that have been commonly available on storage arrays for years. Features like deduplication and compression, thin-provisioning, snapshots, fast-restore, clones, etc. Some of these features exist in the cloud, but usually in a rudimentary form. For example, while storage snapshots exist for AWS EBS, restoration is slow.

For organisations used to the rich data services available on storage arrays, this can feel quite limiting. For many established enterprises, abandoning these features can feel like a dangerous amount of change, as well as a backwards step.

Now, one solution is to re-architect applications to work in a more cloud-native way, and that's generally our advice if you want to really maximize the benefits of using cloud. But this isn't practical for every workload in an enterprise for a variety of non-technical reasons like risk and cost.

Which leaves customers in a bind: some of what cloud offers better suits what they want to do. Ephemeral test/dev workloads, for example, where new code is pushed and runs through a test suite in parallel on infrastructure that would be cost-prohibitive to purchase for permanent deployment on-site, but is easy to spin-up and tear-down in the cloud. But providing access to a clone of a large dataset it needs takes too long using the tools available in the cloud. A co-located storage array with data services accessed over a private data link is one option, but then you're bandwidth and latency constrained.

What the customer wants is a storage array, but in the cloud. Pure Cloud Block Store provides a solution for something the public cloud doesn't offer today, and does it in a way that takes advantage of what the cloud is really about.

04

CONCLUSION

Pure Cloud Block Store represents a good choice for enterprise customers who want to take advantage of public cloud's elasticity and ephemerality but find raw cloud services too costly, particularly at scale, and lacking in performance and features. The choice becomes even clearer if the customer already has some Pure Storage and wants a consistent operational experience using familiar tools and processes. Pure's Evergreen pricing model removes almost any doubt in our minds that Pure customers thinking about using Pure Cloud Block Store should simply try it to see, given the clear benefits and low risk of running a trial.

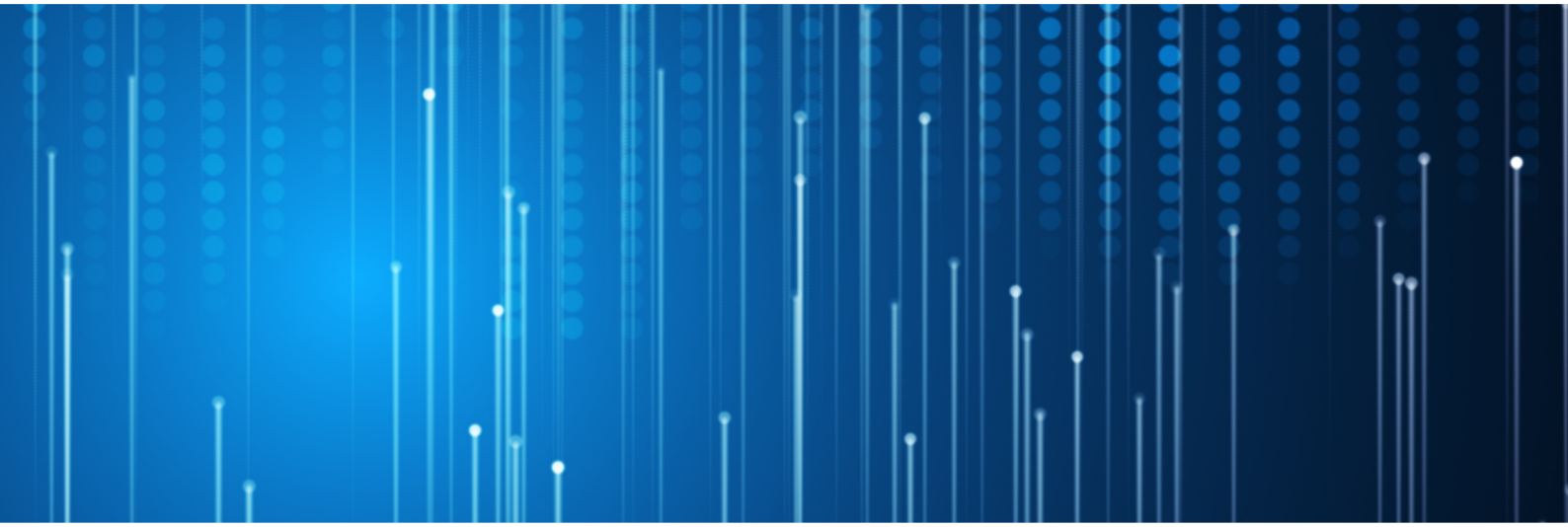
The challenge for Pure is that public clouds are gradually adding more data services to their offerings, particularly as they've moved from being focused on developers building new things to enterprises with existing workloads and lots more money. Compatibility with existing ways of working is necessary to attract these customers because, as discussed above, change is hard.

Pure needs to maintain enough value in its offering for it to be compelling as an alternative to the native cloud service, which creates a strong incentive that benefits customers.

Pure also needs to appeal to Azure customers because its target is the enterprise market, not just the enterprise market that exclusively uses AWS. A compatible service that runs on a given cloud reduces friction for customers wanting to change and if Pure Storage is successful, it sells more of the underlying raw cloud infrastructure anyway. Pure Cloud Block Store provides a migration path from on-site infrastructure to the cloud, so the net growth in cloud is worth supporting Pure as a partner.

PivotNine's view is that this is not insurmountable for Pure, particularly given the different focus and goals of the public clouds and Pure. The public clouds aim to serve a wide variety of customers with something "good enough" for most of them. Pure Storage, by contrast, is not trying to sell Pure Cloud Block Store to everyone with a single terabyte of storage.

Different customers need different things, and that's okay. Pure Cloud Block Store is a good match for the customers it's designed for, which should be the aim of any product worth paying for.



PN

About The Author

Justin Warren is Founder and Chief Analyst of PivotNine. He has worked with many well-known companies around the world, including ANZ, Australia Post, IBM, NetApp, Nutanix, Pure Storage, Red Hat, Suncorp, Telstra, and VMware as well as a variety of startups including Atomist, CodeSee, Cloudistics, Datera, Elastifile, Env0, Habrdata, Hasura, Illumio, Isovalent, Manifold, Mattermost, Smallstep, Spectro Cloud, and Solo.io, among others.

Justin has written for a variety of mastheads, including *Forbes.com*, *iTnews*, *CRN Australia* and *The Saturday Paper*.

Justin holds an MBA from Melbourne Business School, and is a graduate member of the Australian Institute of Company Directors.

About PivotNine

PivotNine Pty Ltd is a specialist IT consulting firm based in Melbourne, Australia.

PivotNine helps customers to evaluate and select technology products, and to implement effective organisational structures and processes.

PivotNine assists vendors with marketing positioning and messaging, with a focus on data driven marketing methods.

Contact

enquiries@pivotnine.com
https://pivotnine.com