

Contrasting Pure & NetApp's Philosophies on Data Reduction

ADMINISTERING DATA REDUCTION

NETAPP'S ON/OFF DATA REDUCTION

1. Ensure free space exists for dedupe metadata
2. Set Inline Compression to ON / OFF
3. (Optional) Choose Adaptive or Secondary Compression
4. Set Inline Dedupe to ON / OFF
5. Set Cross-Volume Dedupe to ON / OFF
6. Set Background Compression (does not run on AFF platforms)
7. Set Background Dedupe
8. Create Efficiency Policy operations
9. Select Schedule Type (Time duration, Manual, New data written threshold)
10. Can schedule max of 8 dedupe or compression operations concurrently per node
11. Avoid overlaps with snapshot/clone operations schedules
12. Continue to ensure sufficient free space exists for metadata

FOR EACH VOLUME / AGGREGATE...

PURE'S ALWAYS-ON DATA REDUCTION

THIS SIDE INTENTIONALLY LEFT BLANK

Data reduction (dedupe + compression) is a massive TCO driver on all-flash arrays since it reduces the amount of raw flash required to store your data. It slashes upfront spend and reduces future spend. For example, consider the savings that average 5:1 data reduction delivers on the Pure Storage FlashArray: you purchase only 35TBs raw to get 100TBs effective capacity. In contrast, 125TBs raw are typically required on legacy, non-data reducing arrays. That's 90TBs in savings for every 100TBs effective. Dramatic savings that lower CapEx and OpEx (lower power/cooling/rack units) and make all-flash arrays affordable for general purpose workloads, not just the most performance intensive workloads. Pure pioneered the use of data reduction in its unique recipe for all-flash arrays. And like everything on Pure, it's delivered with simplicity. Our data reduction is Always-On and every customer can benefit from the efficiency savings without any tuning or performance impacts. We've done the heavy lifting to make this feat possible so you don't have to spend any time thinking about or managing it - you just get the benefits without any trade-offs.

Over the last couple of years, NetApp, Dell-EMC, and HPE have taken a different approach to data reduction. They have added (bolt-on) dedupe, compression, or both on top of their decades-old, retrofit all-flash arrays with a key distinction: data reduction is switched On/Off manually, not Always-On. **These legacy vendors claim that On/Off data reduction delivers efficiency with greater flexibility. Is this true? Or, is it creative spin for forcing you to decide between performance and efficiency - while adding tons of complexity?**

This blog makes the case that it's the latter. NetApp all-flash arrays are a great example, though the inherent tradeoffs in performance vs. efficiency vs. simplicity apply equally well to all retrofit all-flash arrays. These tradeoffs require customers to either forego configuring data reduction or limit the use to non-critical workloads, thereby sacrificing the majority, if not all, of potential data reduction savings. By

contrast, Pure customers typically benefit from 2X better data reduction through industry leading, Always-On data reduction natively engineered into Pure Cloud-Era flash arrays. Read-on to learn more.

Why Isn't Data Reduction Just Always-On?

In the cloud era, simplicity thru minimizing (and ideally eliminating) knobs and tuning is a critical enabler of automation and agility. If data reduction worked with zero impact to performance, you would want and expect it to be Always-On. But, it turns out, that's not the case on retrofit all-flash arrays.

Here's an excerpt from [NetApp's OnTap9 documentation](#) that provides a carefully worded hint at the very real performance impact:

"Adaptive compression is preferred when there are a high volume of random reads on the system and higher performance is required. Secondary compression is preferred when data is written sequentially and higher compression savings are required."

What happens if you configure "secondary compression" and the data workload transitions over time to high volume of random reads? But even before that, customers are forced to figure out the characteristics of their workloads, which is wasted time and effort in the cloud era. More on that below.

To add to your confusion, quoted performance specs on the retrofit all-flash arrays do not include data reduction, or you'd expect vendors to tout that. That leaves you in the dark about the "real-world" performance of these arrays once data reduction is switched On.

Data reduction on the Pure FlashArray and FlashBlade is Always-On and the performance specs include the benefits of data reduction. As a result, you can be confident you'll get the effective storage you expect and the performance your applications deserve.

How do you know in advance whether dedupe, compression, or both should be ON?

Sure, there are rules of thumb - such as dedupe works best for virtual servers and virtual desktops and compression for databases - but, these can be tricky to apply. Consider the following examples: How would this work for cloud service providers or larger enterprises who typically don't know which workloads their customers will deploy? While compression is typically beneficial for databases, dedupe also delivers savings since multiple copies of the same database often exists. The only real way to find out, without leaving real savings on the table, is to turn ON dedupe and compression.

The inherent gamble for you - the admin - is that if you suspect switching ON data reduction features will have an unknown performance impact, would you knowingly take that risk with your production workloads? In our experience most rational folks sacrifice the efficiency savings in favor of not adding performance risk to their applications. And that's understandable - but obviously not desirable.

What magnifies this risk is that **if the performance impact with data reduction switched On turns out to be intolerable, you can't just simply turn it Off and return back to pre-data reduction state immediately.** Data that had been reduced will have to be re-hydrated, which means additional performance overhead/penalty until the re-hydration is complete. There are additional (more dangerous) implications of turning off data reduction. The volume must have enough space available to store the non-reduced data. Running out of space is not a situation that you want to get to. Perhaps this is one of the

many reasons why [NetApp's OnTap9 documentation](#) requires that you first involve technical support:

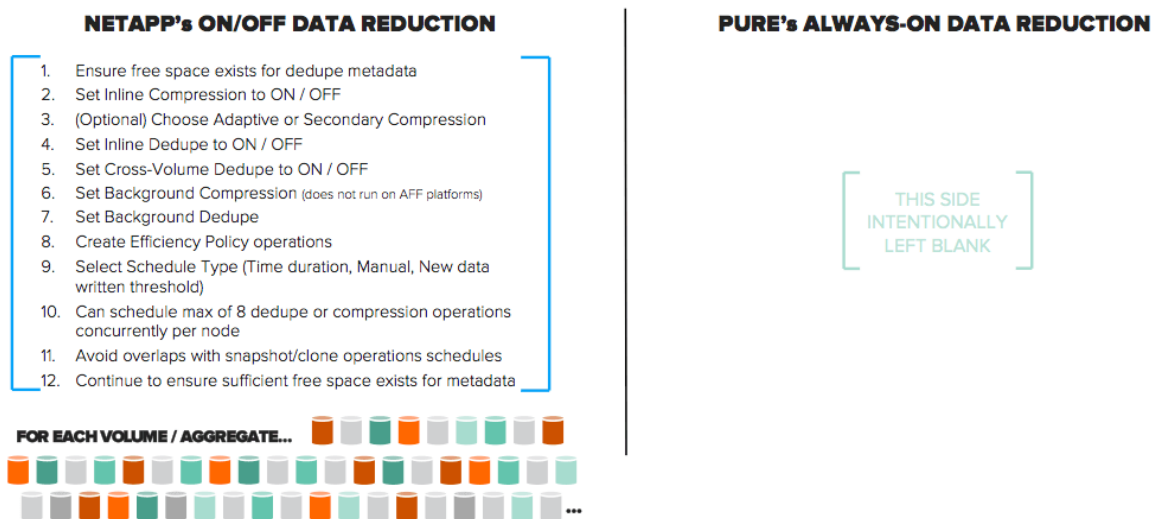
*"You can choose to remove the space savings achieved by running efficiency operations on a volume. **You must ensure that you contact technical support before removing or undoing the space savings on a volume.**"*

With Always-On data reduction, you automatically get the benefit of dedupe and/or compression without worrying about pre-planning data reduction settings by workload or risking the complex back-out of settings that require support involvement.

How complex is it to manage dedupe, compression, or both manually?

Simply put: VERY! Here's an illustration of the potential steps involved in administering data reduction on the NetApp AFF A-Series. This is our understanding based on a review of NetApp product documentation.

ADMINISTERING DATA REDUCTION



All in all, there are about a dozen steps involved. Your journey of configuring data reduction on NetApp begins with ensuring there is sufficient space for dedupe metadata at the Aggregate and FlexVol levels. While inline dedupe and compression will default to On on the AFF A-Series, you will want to still decide whether to leave it On or switch to Off, and whether Adaptive or Secondary compression should be configured, given the aforementioned caveats in this blog. Once you have made these decisions, you need to configure schedule types and orchestrate the schedules such that a max of 8 dedupe or compression operations execute concurrently per node. If you plan to use snapshots for data protection, you should be aware that bolt-on dedup can't operate on data once it is protected in a snap, and so, you also must ensure that background dedup operations regularly complete before any snapshots are taken. Finally, you need to continue to ensure sufficient free space remains for dedupe metadata.

Complicating this further is the granularity at which these features have to be configured. Because dedupe and compression operate at the Aggregate and FlexVol level, you need to repeat these steps for potentially hundreds of Aggregates and FlexVols. And you'll need to revisit them as the characteristics of your workloads change (and new ones get added) over time. Even if you are willing to live with the complexity,

the pitfall of managing this complexity at scale is that errors are bound to occur and be compounded over time, leading to unintended consequences.

Does this sound like fun? We don't think so either. Fortunately, you can completely avoid all this complexity with Pure's Always-On data reduction.

What are the overall implications of On/Off data reduction?

The unknown performance tradeoffs and complexity have a direct effect on the effective capacity that you can achieve and the TCO of your storage purchase. Given the tradeoffs, it's likely that you may entirely forego switching On data reduction on your array, or severely limit its use to volumes supporting non-critical workloads, thereby sacrificing the majority, if not all, of the efficiency savings from data reduction. Not only does this increase the TCO, it leaves you to face an undesired outcome: effective capacity shortfall. The quoted effective capacity will have assumed the benefit of data reduction, and now the realized effective capacity without data reduction is far less. Of course, you can negotiate with the legacy vendor to provide free capacity to make up for the shortfall, but the vendor is likely to recommend enabling data reduction as the first step, pointing you back into the dilemma of performance vs. efficiency tradeoff that you were trying to avoid in the first place.

What Questions Should You Ask Vendors?

If you find yourself in the market for an all-flash array, here are a few questions that can help you better understand the implications of On/Off data reduction approach on the array:

- 1) Why isn't dedupe/compression Always-On?
- 2) What is the performance impact (at scale) of turning On dedupe/compression?
- 3) What is the performance impact and gotchas of turning Off dedupe/compression?
- 4) What are the steps involved in switching On or Off dedupe/compression?
- 5) How will I know up front that a given workload will benefit from dedupe, compression, or both?
- 6) How will I know if and when I need to re-evaluate my dedupe/compression for the workloads on my array?
- 7) What (if any) are the dependencies of dedupe/compression schedules vs. snapshot schedules?
- 8) How many max dedupe/compression operations can execute in parallel?
- 9) What are the flash wear implications of switching On or Off dedupe/compression?

Summary

Data reduction holds great promise for the efficiency savings that it offers, which lower your upfront spend and slow future spend. While retrofit all-flash array vendors will market On/Off data reduction as "flexibility", they are forcing you to choose between performance, efficiency and simplicity. With Always-On data reduction on the Pure Cloud-Era flash arrays, you can get all three without tradeoffs.