



# **Determining the array LUNs for specific aggregates**

## **ONTAP FlexArray**

NetApp  
February 11, 2024

This PDF was generated from [https://docs.netapp.com/us-en/ontap-flexarray/install/concept\\_rules\\_for\\_mixing\\_storage\\_in\\_aggregates\\_for\\_v\\_series\\_systems.html](https://docs.netapp.com/us-en/ontap-flexarray/install/concept_rules_for_mixing_storage_in_aggregates_for_v_series_systems.html) on February 11, 2024. Always check docs.netapp.com for the latest.

# Table of Contents

- Determining the array LUNs for specific aggregates ..... 1
  - Rules for mixing storage in array LUN aggregates ..... 1
  - How the checksum type is determined for array LUN aggregates..... 1
  - Checksum type considerations for adding spare array LUNs to aggregates..... 2
  - Aggregate rules when the storage arrays are from the same family ..... 2
  - Aggregate rules when the storage arrays are from different vendors or families..... 3

# Determining the array LUNs for specific aggregates

There are a number of rules about mixing different types of storage in aggregates that are unique to ONTAP systems that use array LUNs. You must understand these rules when planning which array LUNs and disks to add to which aggregates.

## Rules for mixing storage in array LUN aggregates

When planning for aggregates, you must consider the rules for mixing storage in aggregates. You cannot mix different storage types or array LUNs from different vendors or vendor families in the same aggregate.

Adding the following to the same aggregate is not supported:

- Array LUNs and disks
- Array LUNs with different checksum types
- Array LUNs from different drive types (for example, FC and SATA) or different speeds
- Array LUNs from different storage array vendors
- Array LUNs from different storage array model families



Storage arrays in the same family share the same performance and failover characteristics. For example, members of the same family all perform active-active failover, or they all perform active-passive failover. More than one factor might be used to determine storage array families. For example, storage arrays with different architectures would be in different families even though other characteristics might be the same.

## How the checksum type is determined for array LUN aggregates

Each ONTAP aggregate has a checksum type associated with it. The aggregate checksum type is determined by the checksum type of the array LUNs that are added to it.

The checksum type of an aggregate is determined by the checksum type of the first array LUN that is added to the aggregate. The checksum type applies to an entire aggregate (that is, to all volumes in the aggregate). Mixing array LUNs of different checksum types in an aggregate is not supported.

- An array LUN of type *block* must be used with block checksum type aggregates.
- An array LUN of type *zoned* must be used with advanced zoned checksum (AZCS or advanced\_zoned) type aggregates.

Before you add array LUNs to an aggregate, you must know the checksum type of the LUNs you want to add, for the following reasons:

- You cannot add array LUNs of different checksum types to the same aggregate.

- You cannot convert an aggregate from one checksum type to the other.

When you create an aggregate you can specify the number of array LUNs to be added, or you can specify the names of the LUNs to be added. If you want to specify a number of array LUNs to be added to the aggregate, the same number or more array LUNs of that checksum type must be available.

## Checksum type considerations for adding spare array LUNs to aggregates

You must account for certain considerations related to checksum types when adding spare array LUNs to aggregates. For example, if you plan to add a spare array LUN to an aggregate by specifying its name, you must ensure that array LUN and the aggregate have the same checksum type.

The following are some checksum type considerations for adding spare array LUNs to aggregates:

- You cannot mix array LUNs of different checksum types in an array LUN aggregate.
- If you specify a number of spare array LUNs to be added to an aggregate, ONTAP by default selects the array LUNs of the same checksum type as the aggregate.
- Array LUNs of the zoned checksum type when added to an existing zoned checksum aggregate continue to be zoned checksum array LUNs.
- Zoned checksum spare array LUNs added to an Advanced Zone Checksum (AZCS) type aggregate use the AZCS checksum scheme.



You can check the checksum type of the spare array LUNs by using the `storage disk show` command. For more information about the command, see the man pages.

## Aggregate rules when the storage arrays are from the same family

Specific rules apply to how you can lay out array LUNs in aggregates when the storage arrays are from the same storage array vendor and model family.

If your storage arrays are from the same vendor, the rules for adding array LUNs to aggregates are as follows:

- You can mix array LUNs from the storage arrays in the same aggregate if the storage arrays are in the same family.
- You can separate the array LUNs into different aggregates.

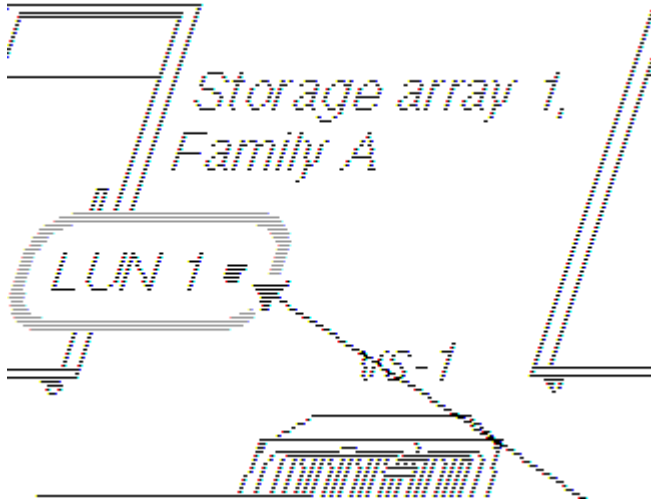
The following examples show some options for laying out array LUNs in aggregates when the storage arrays behind an ONTAP system are in *the same vendor family*.



For simplicity, the illustrations show only two storage arrays; your deployment can include more storage arrays.

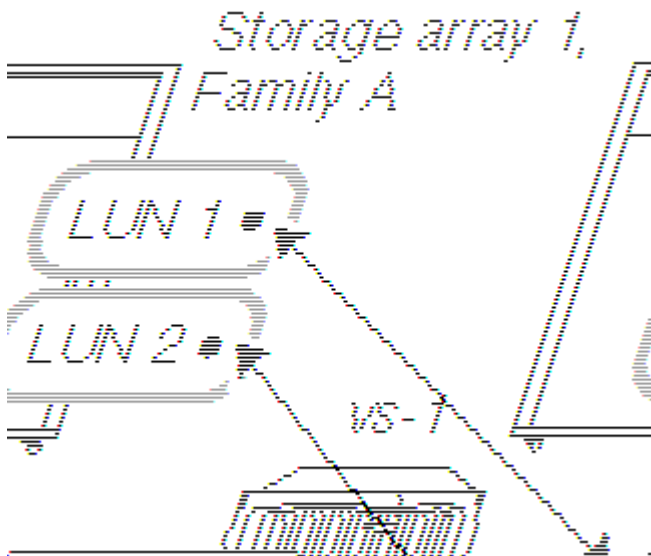
## Example 1: Add LUNs from all storage arrays to a single aggregate

As shown in the following illustration, you can create one aggregate, then add all LUNs from all the storage arrays in the same family to the same aggregate:



## Example 2: Distribute and mix LUNs from the storage arrays over multiple aggregates

As shown in the following illustration, you can create multiple aggregates, then distribute and mix the array LUNs from the different storage arrays in the same family over the aggregates:



This example is not supported if you have storage arrays of the same model and one has Fibre Channel drives and the other storage array has SATA drives. In that case, those storage arrays are not considered to be in the same family.

## Aggregate rules when the storage arrays are from different vendors or families

Specific rules apply to how you can lay out array LUNs in aggregates when the storage

arrays are from different vendors or from different storage array families from the same vendor.

The following rules apply if your storage arrays are from different vendors or different families from the same vendor:

- You cannot mix array LUNs from storage arrays from different vendors, or from different families of the same vendor, in the same aggregate.
- You can associate the aggregate containing the root volume with any of the storage arrays, regardless of the family type of the storage array.

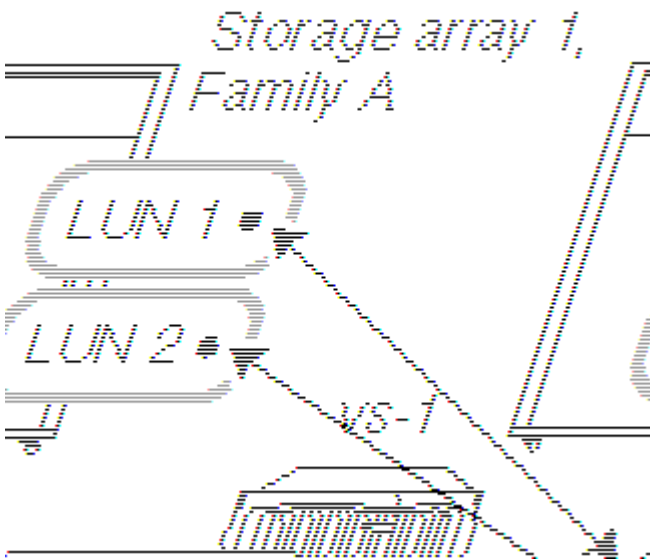


When you create your aggregate, be sure that you explicitly specify the IDs of the array LUNs that you want to add to the aggregate. Do not use the parameters for specifying the number and size of array LUNs to be picked up because the system might automatically pick up LUNs from a different family or from a different vendor's storage array. After array LUNs from different families or vendors are in the same aggregate, the only way to fix the problem of mixed array LUNs in an aggregate is to destroy the aggregate and re-create it.

The following examples show options for how to lay out array LUNs in aggregates when the storage arrays are from *different vendors or from different families from the same vendor*.

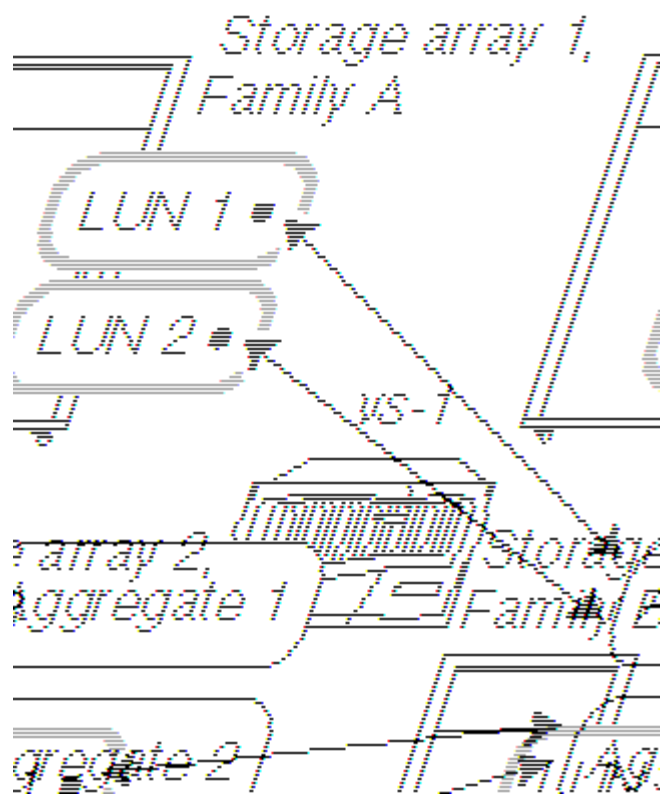
### Example 1: LUNs from the two storage arrays are in different aggregates

In this example, some LUNs for ONTAP are from Storage array 1, Family A, while the other LUNs for ONTAP are from Storage array 2, Family B. The LUNs from the two storage arrays cannot be added to the same aggregate because the two storage arrays are from different families from the same vendor. The same would be true if the two storage arrays were from different vendors.



### Example 2: Some LUNs can be mixed in the same aggregate and some cannot

In this example, one storage array is from Family A and the other two storage arrays are from Family B. The LUNs from the Family A storage array cannot be added to the same aggregate as the LUNs from a Family B storage array because the storage arrays are from different families. However, LUN 1 of storage array 3 can be assigned to aggregate 2, which also contains LUNs from storage array 2, because the two storage arrays are in the same family.



## Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.