NetApp

ONTAP Backstage

Wie tickt ONTAP?

Stephan Egle

Data Architect

Michael Spengler

Account Technology Specialist

Da die Zeit sehr knapp ist und wir viel Content rüberbringen wollen bitten wir Fragen nach der Session zu stellen.

Wir stehen Ihnen im Anschluß an die Session in der “Lounge Area” zur Verfügung

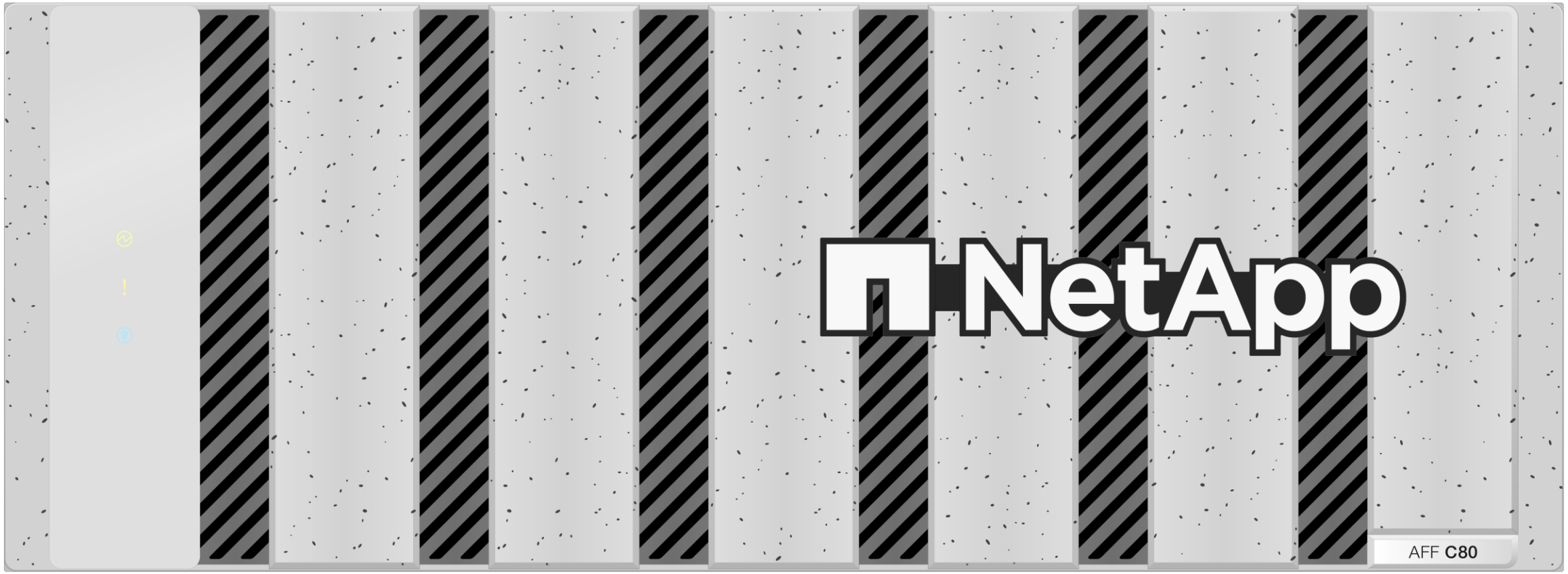
NetApp

Agenda

- ONTAP Architektur
- Read / Write Workflows
- Storage Efficiency (z.B. Inline Dedup, Compression, …)
- WAFL Data Structure & Tetris
- Snapshots & Clones
- Lese / Schreib Pfad Optimierung
- Parallelisierung
- Warum interessiert das eigentlich und was haben sie davon?

ONTAP Architecture

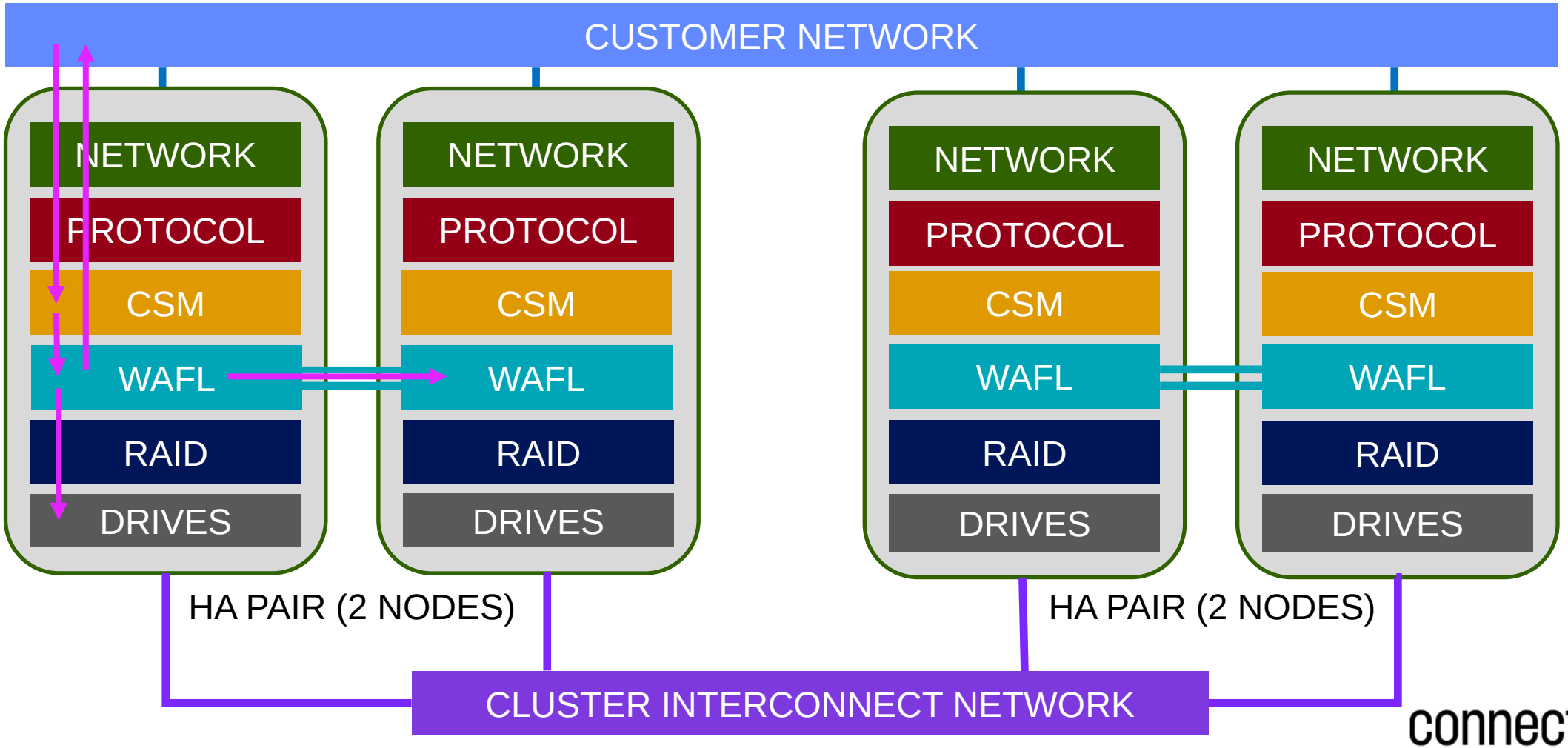
Overview



READ / WRITE WORKFLOWS

DIRECT & INDIRECT

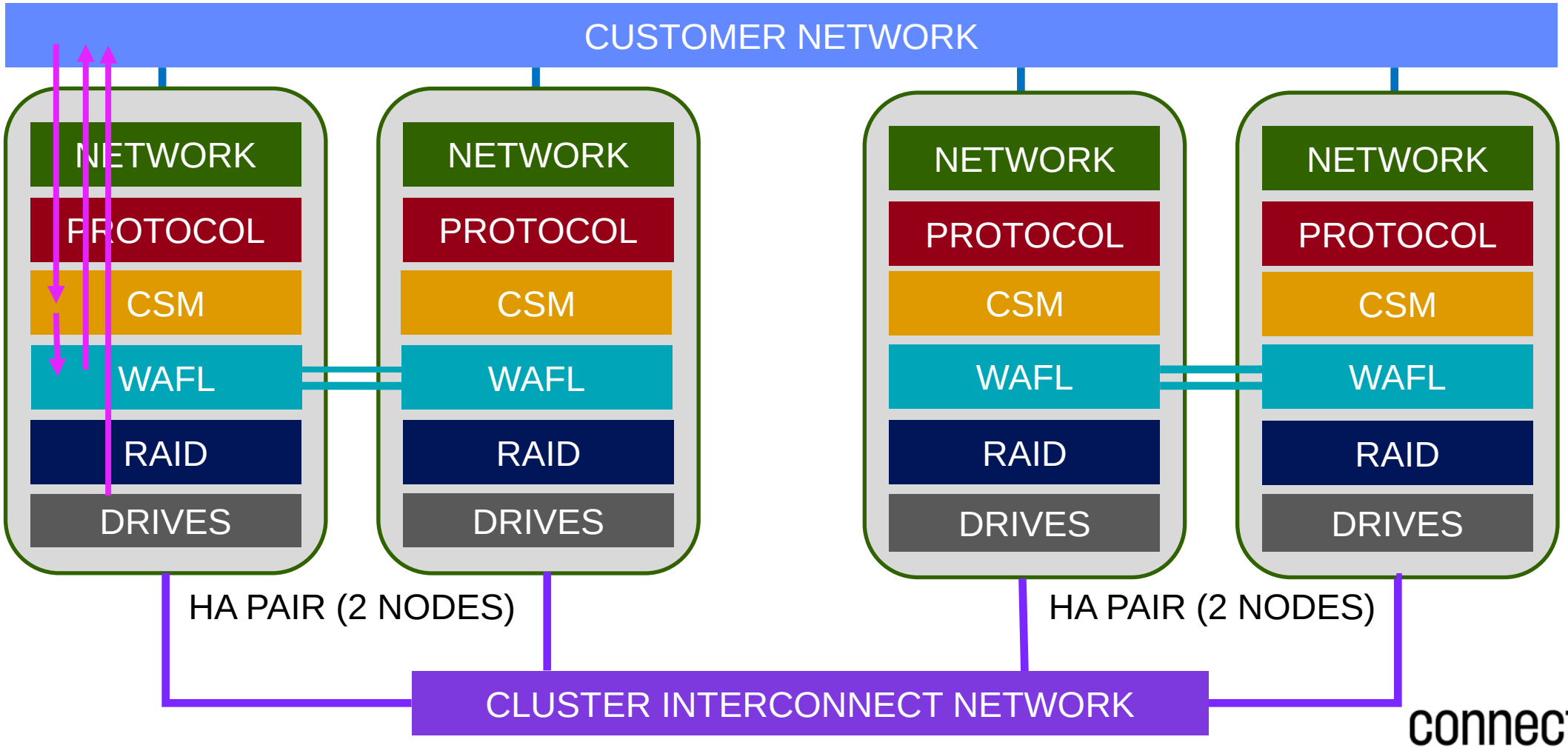
1. DIRECT WRITE



READ / WRITE WORKFLOWS

DIRECT & INDIRECT

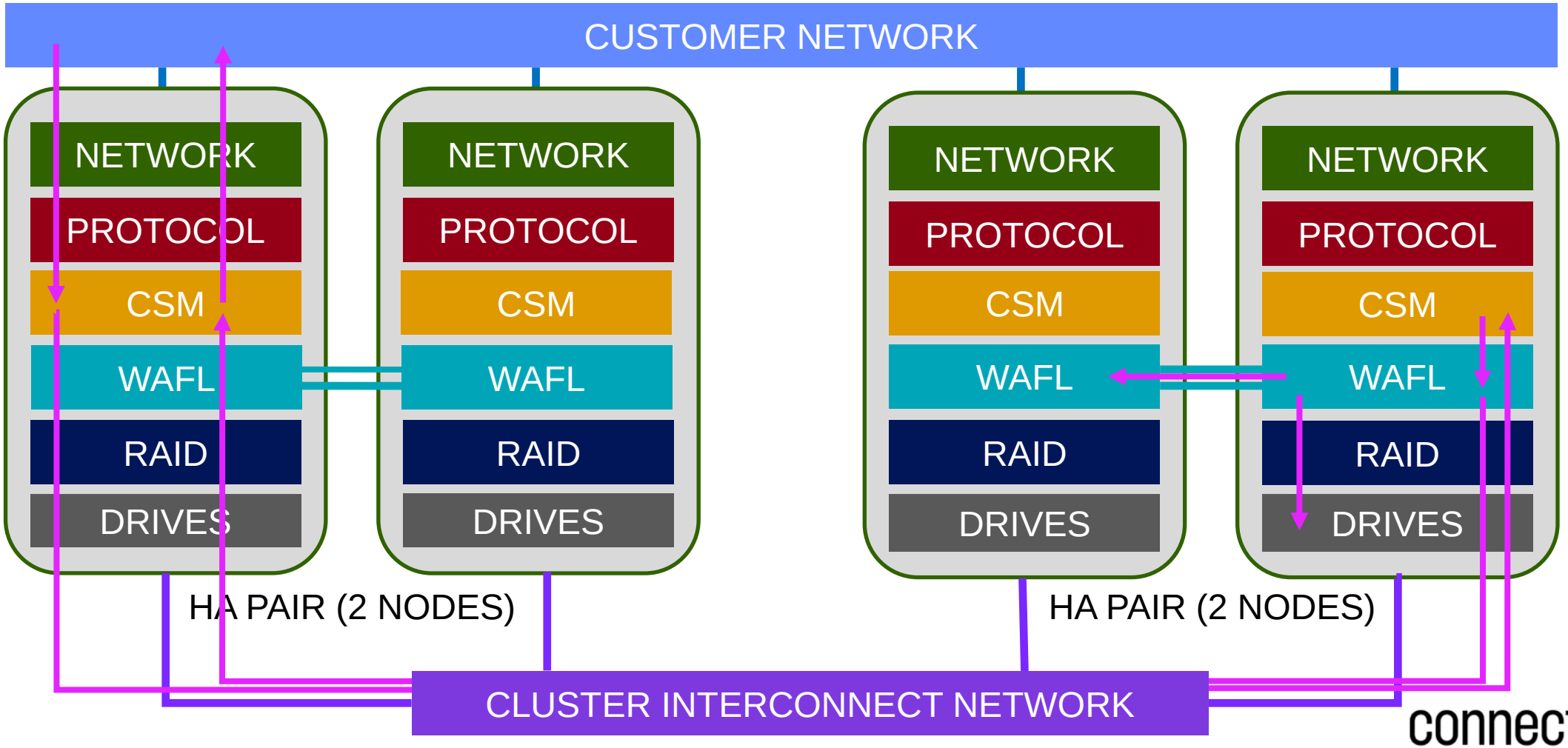
2. DIRECT READ



READ / WRITE WORKFLOWS

DIRECT & INDIRECT

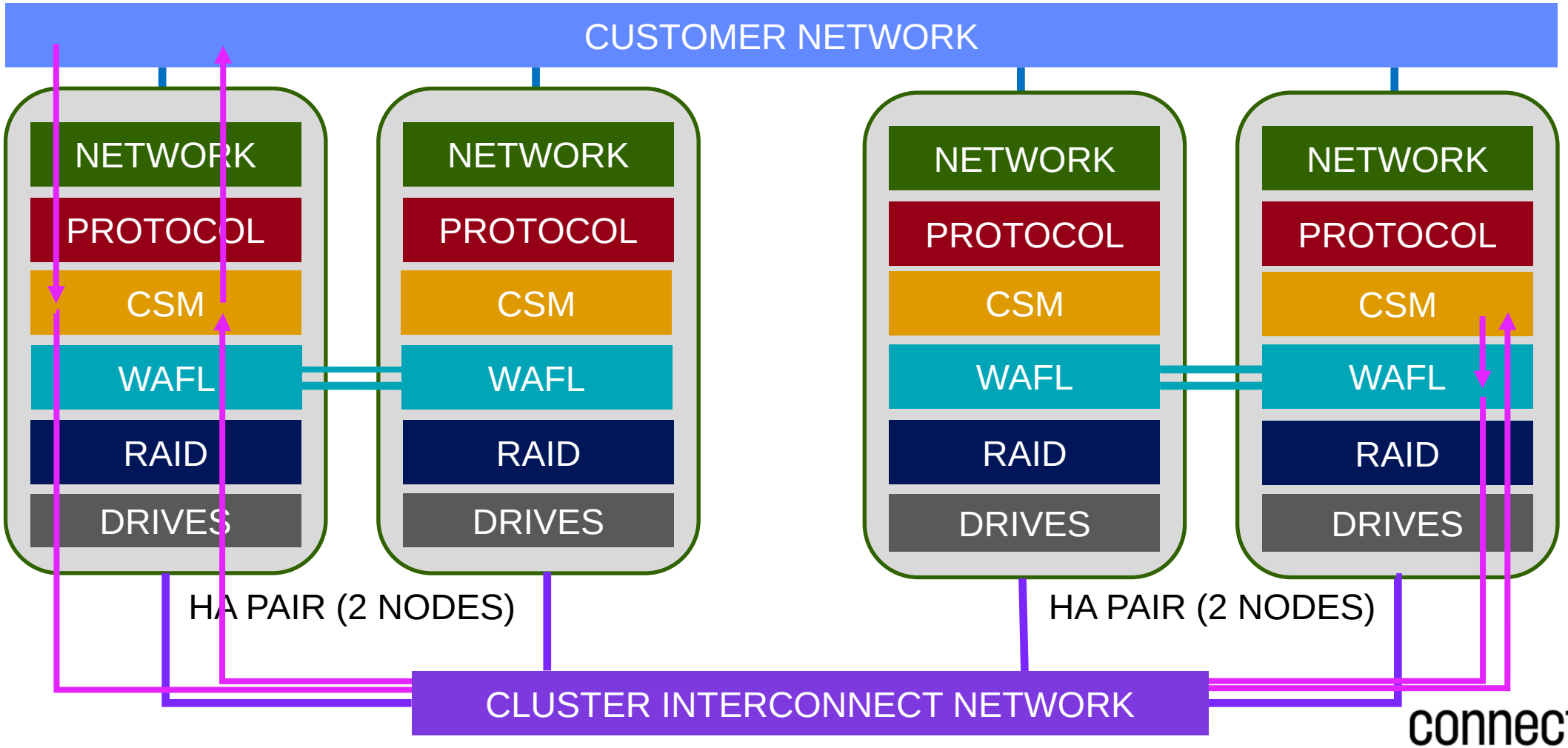
3. INDIRECT WRITE



READ / WRITE WORKFLOWS

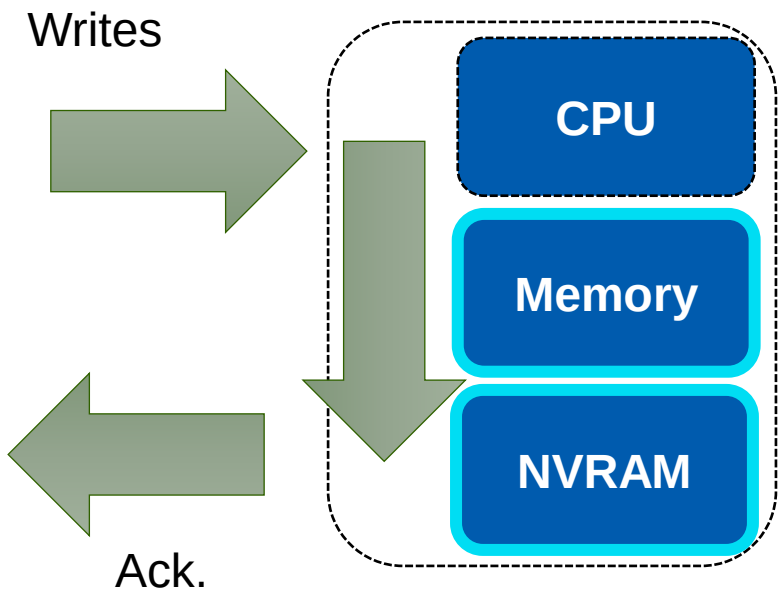
DIRECT & INDIRECT

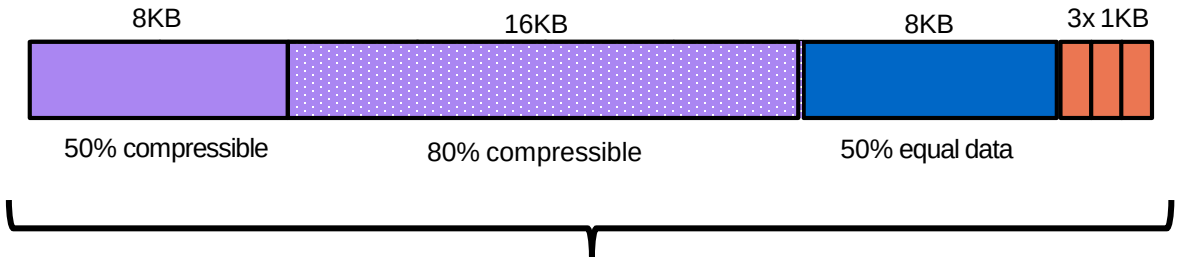
4. INDIRECT READ



Storage Efficiency

WAFL Workflow





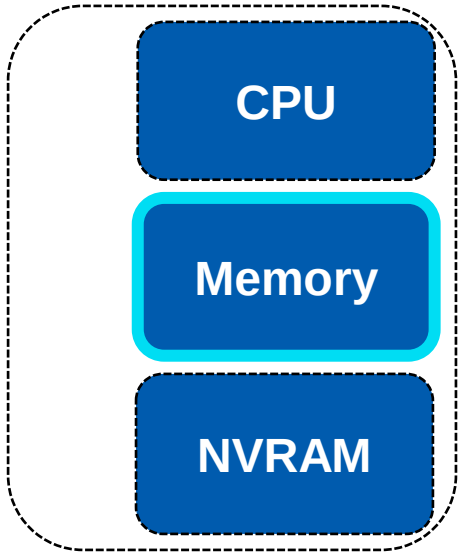
Data set received from client in this example:

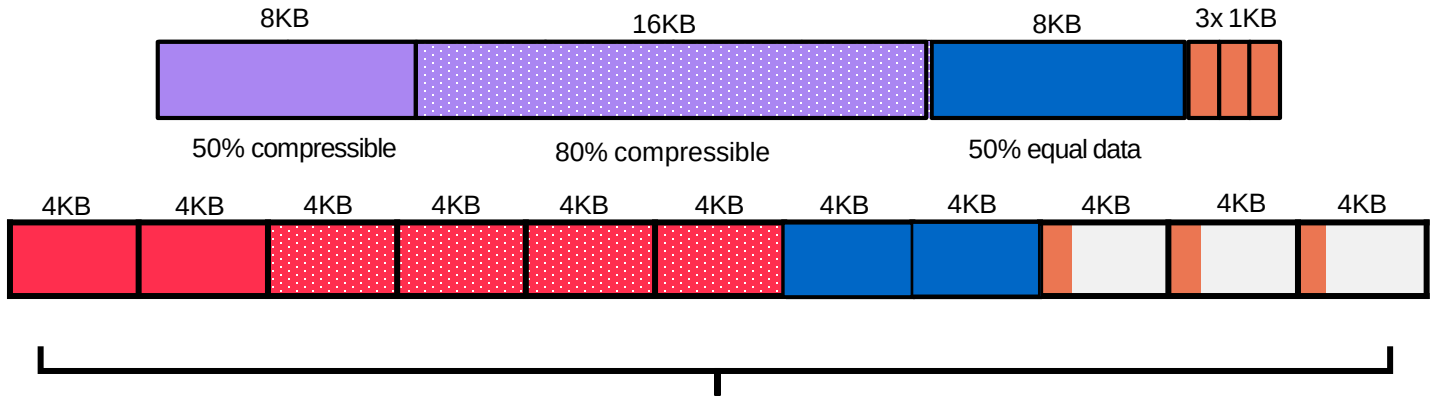
- Files with different size
- Different behavior in deduplication or compression ratio

- Soon as data is moved to NVRAM, acknowledge gets send out to client

Storage Efficiency

WAFL Workflow

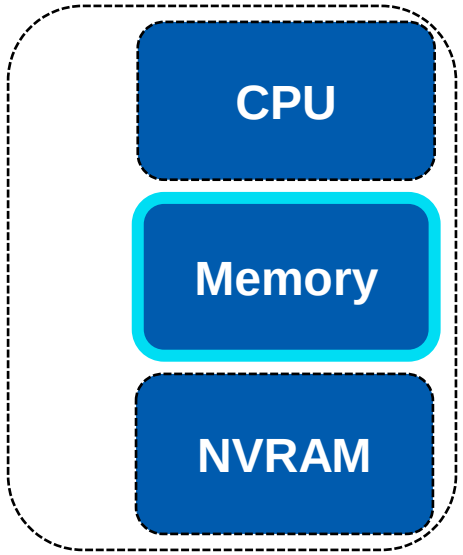


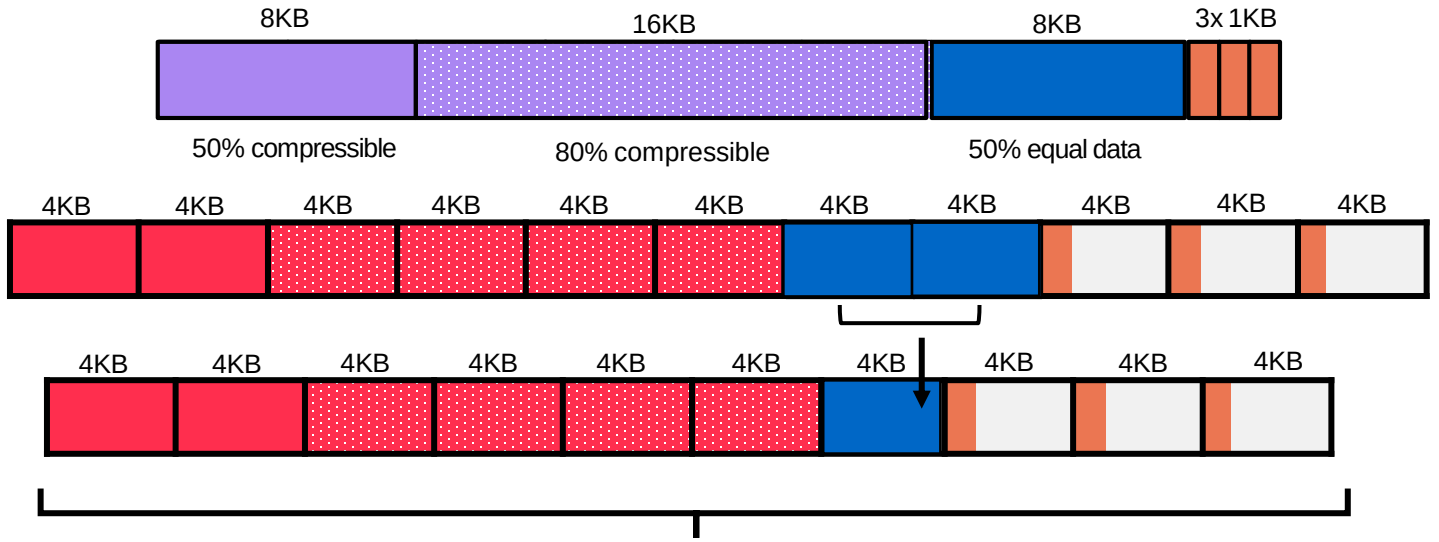


- WAFL divides the data set into 4KB blocks

Storage Efficiency

WAFL Workflow





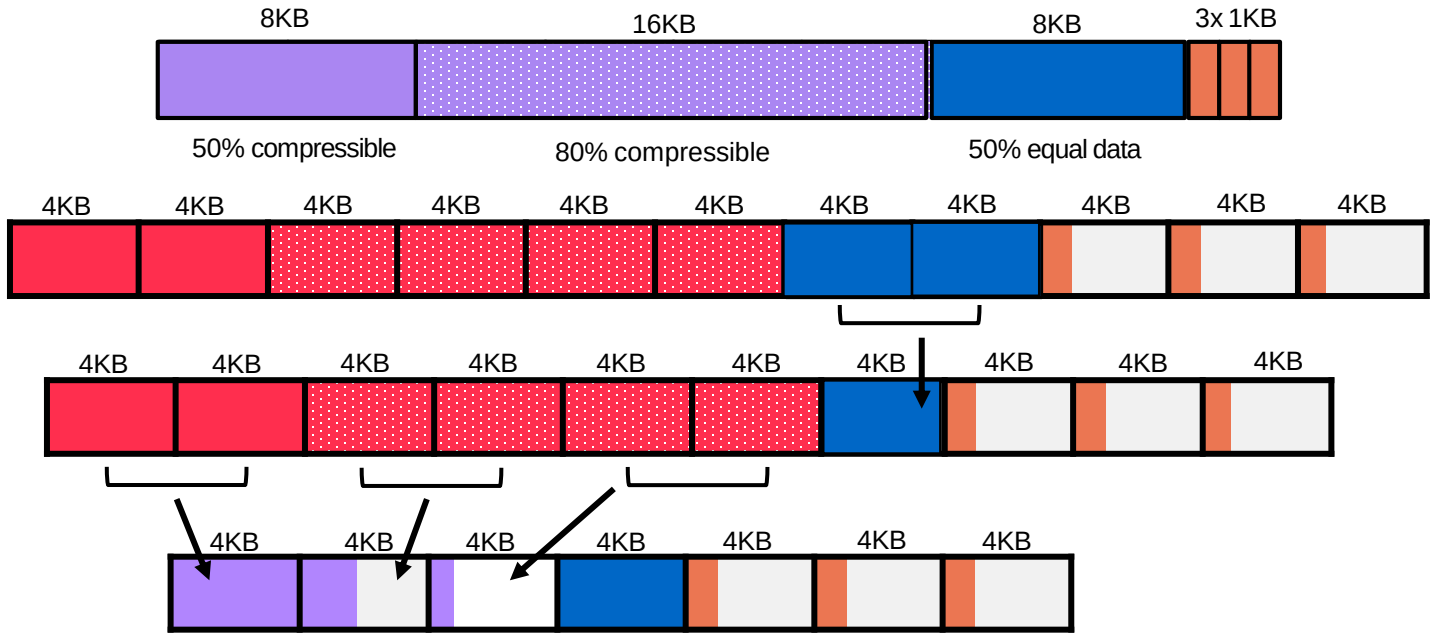
First Storage Efficiency feature to run in memory:

- Zero Pattern Detection + Inline Deduplication
- Duplicate data get detected and repointered
- Based on 4KB fingerprints

Storage Efficiency

WAFL Workflow

- Inline Compression builds 8KB compression groups and tries to compress the data set

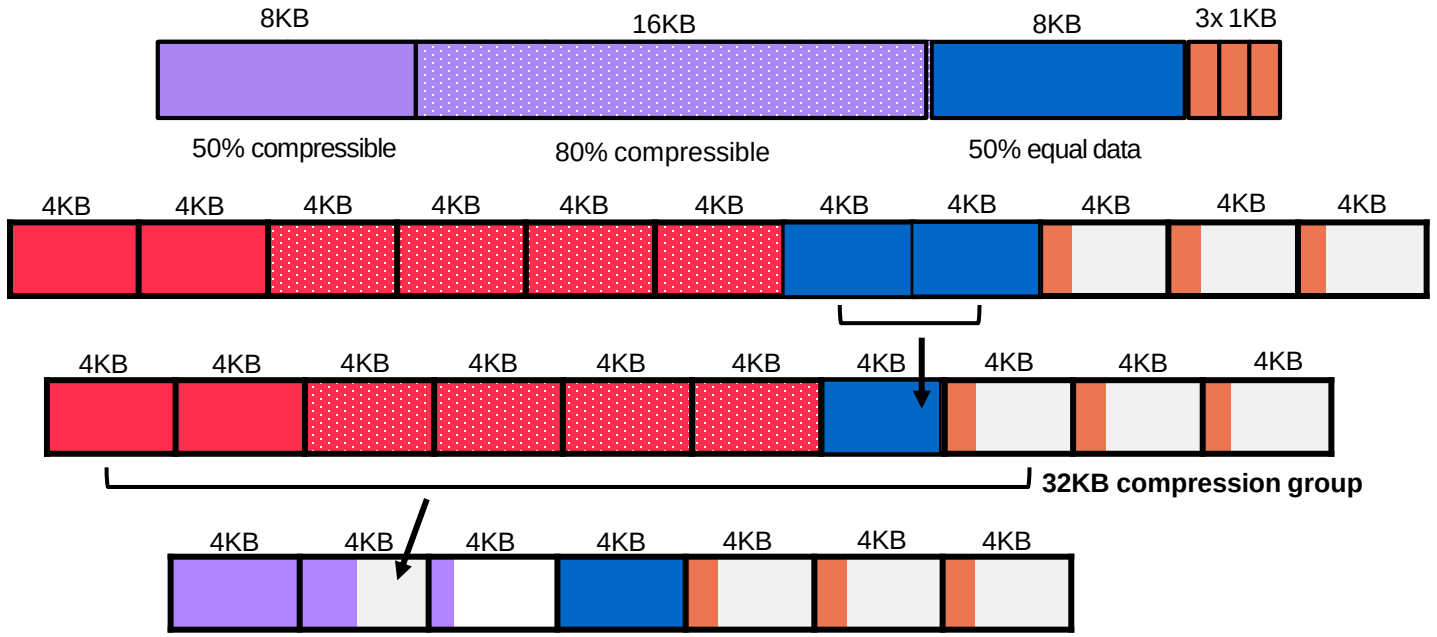


Storage Efficiency

WAFL Workflow

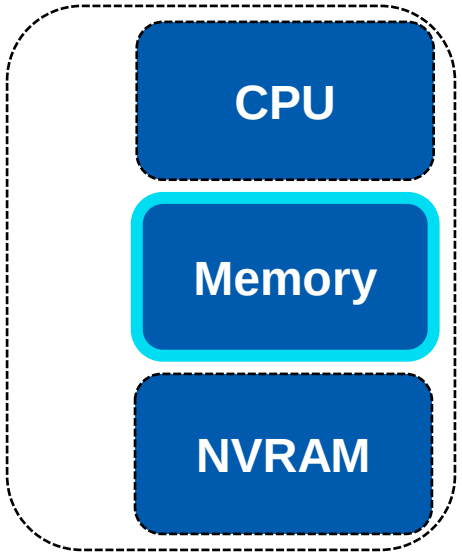
- Inline Compression builds 8KB compression groups and tries to compress the data set
- Since ONTAP version 9.15 inline compression groups can scale up to 32KB compression groups (limited to hardware)

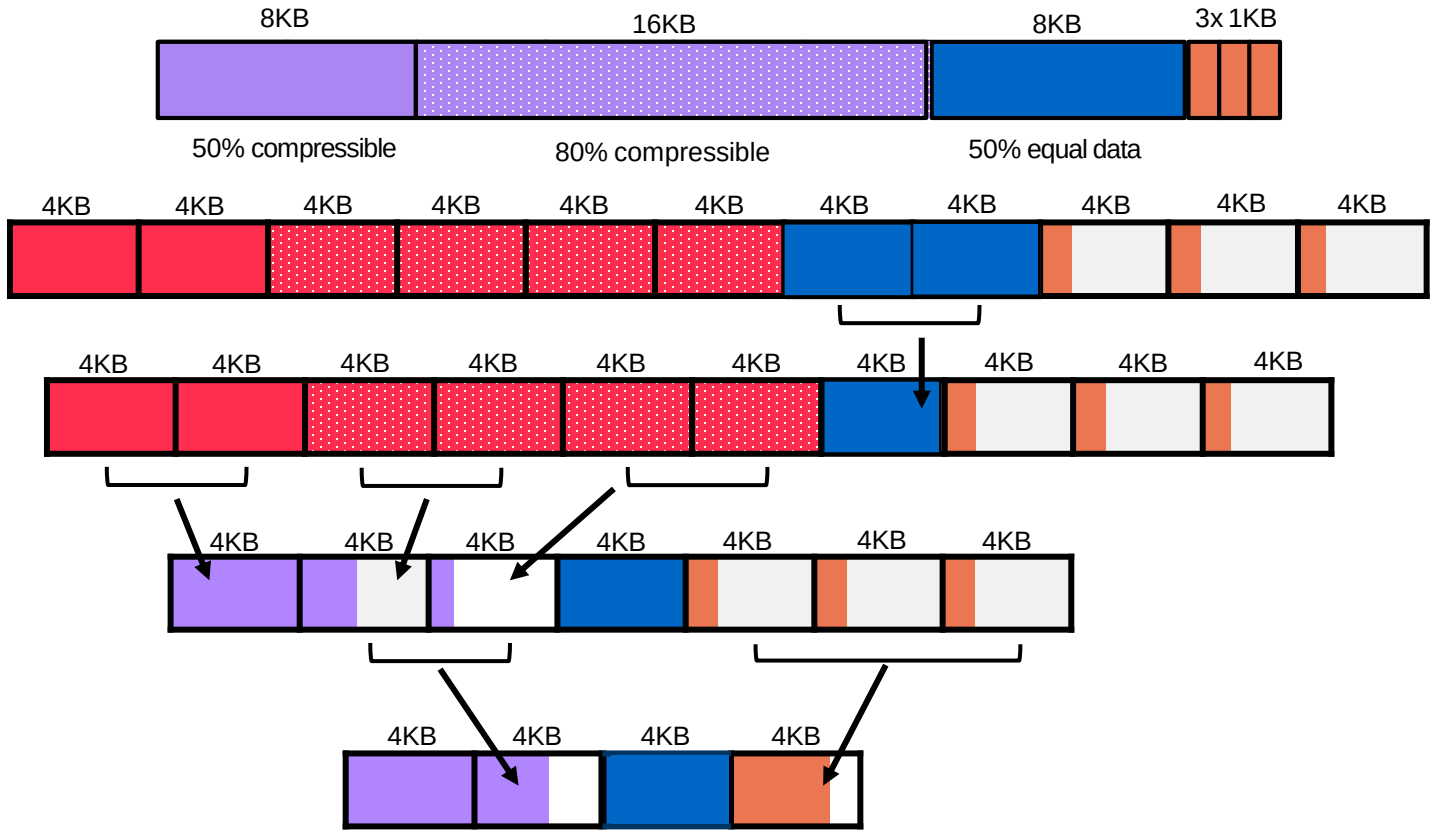
- Background compression always uses 32KB compression groups



Storage Efficiency

WAFL Workflow

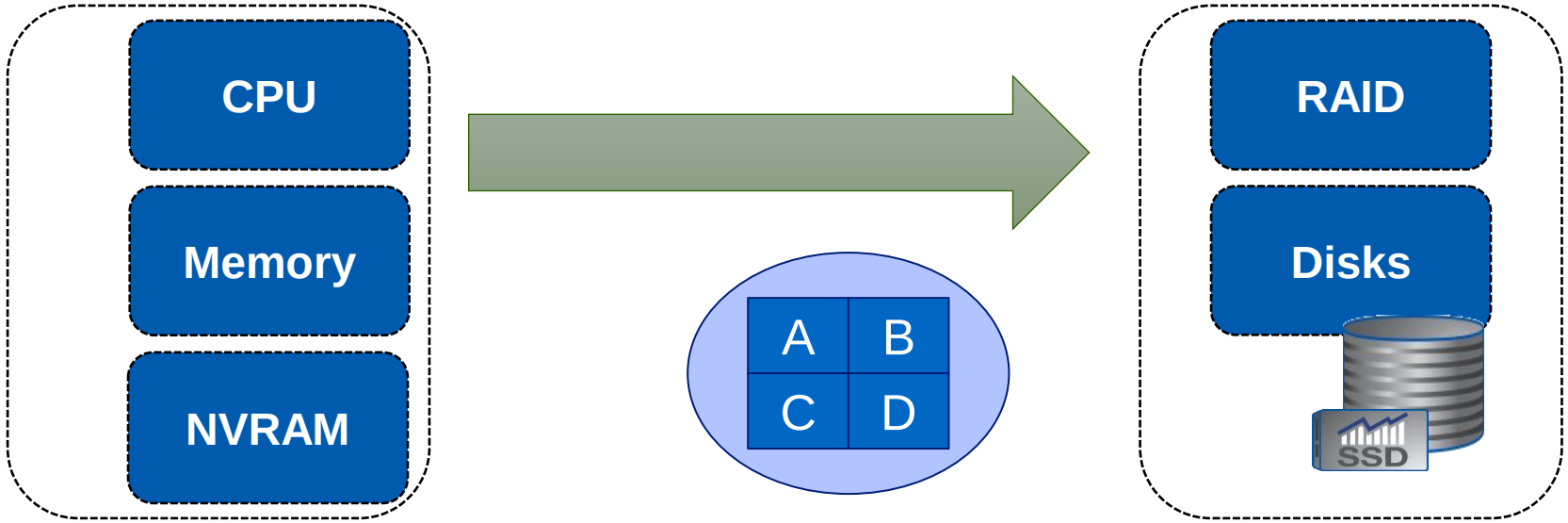




- Inline Compaction merges data from blocks with unused capacity

Storage Efficiency

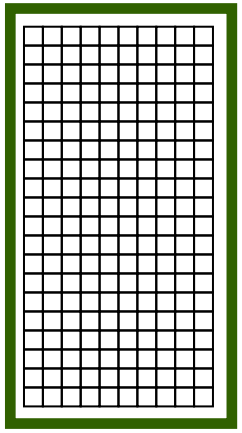
WAFL Workflow



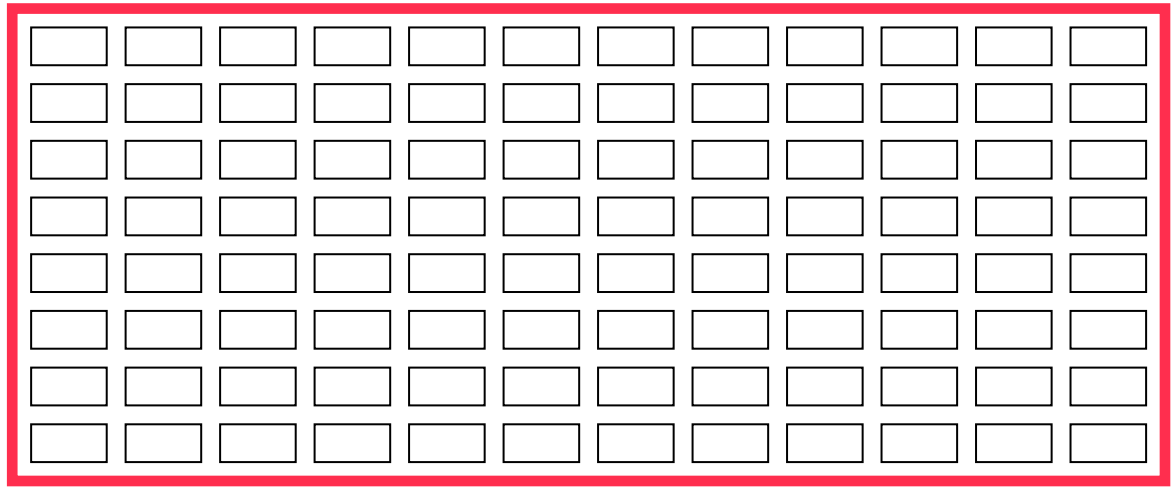
- After Efficiency features have been applied, data is moved to disk

WAFL Data Structures

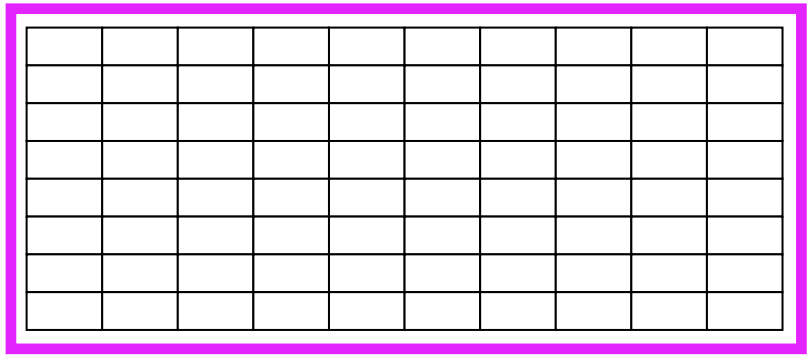
Free Block Bitmap

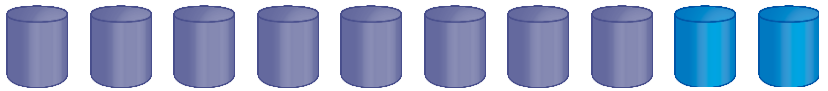


WAFL Cache



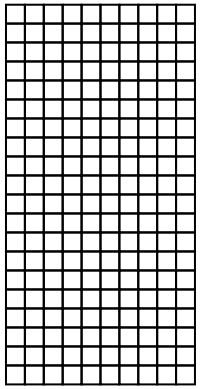
Tetris



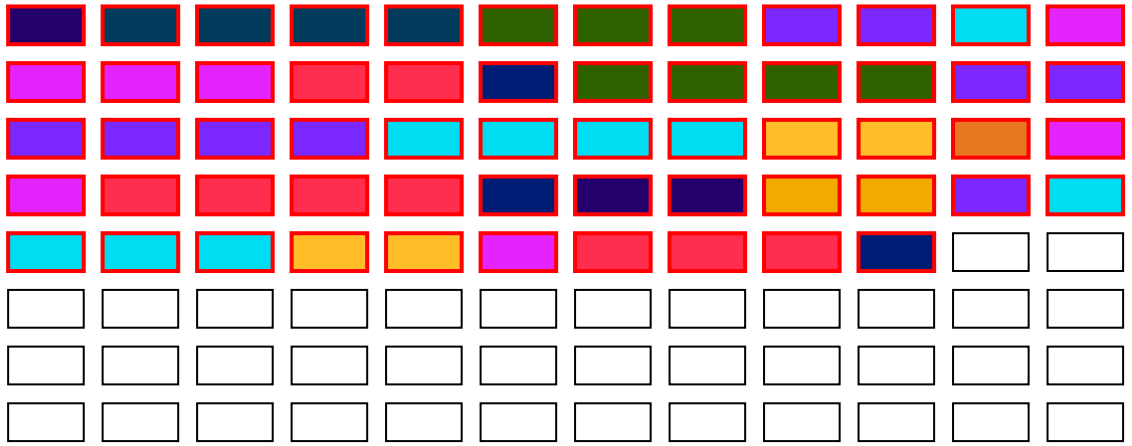


WAFL Write Cache

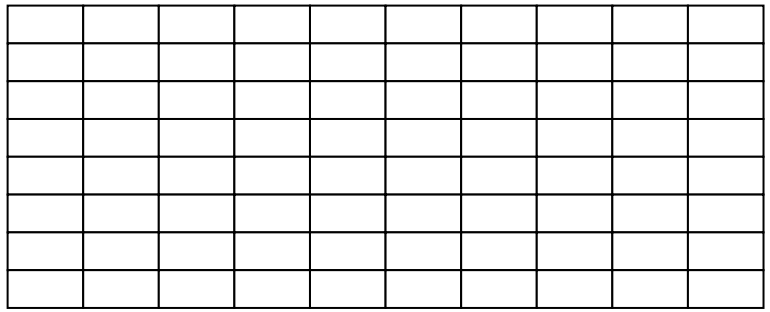
Free Block Bitmap

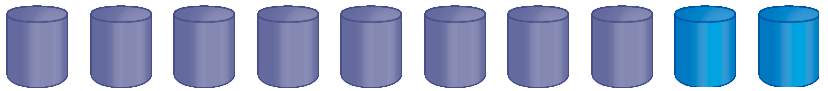


WAFL Cache



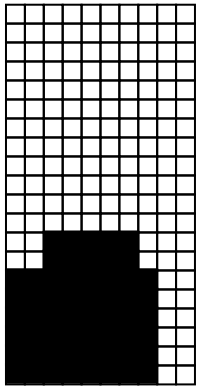
Tetris



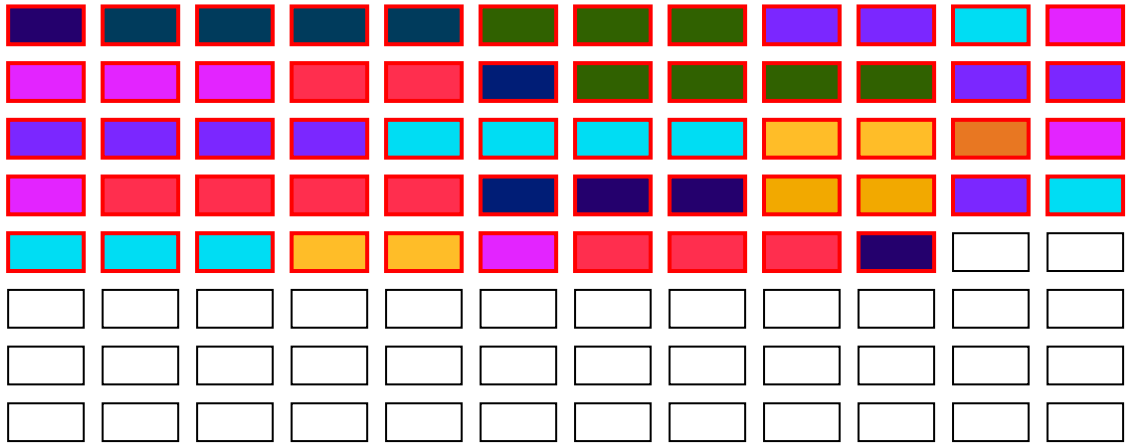


WAFL Tetris

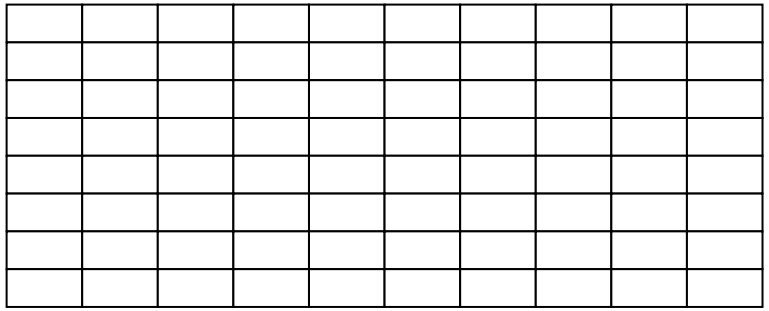
Free Block Bitmap

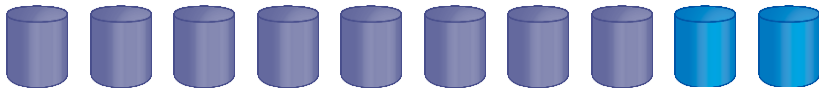


WAFL Cache



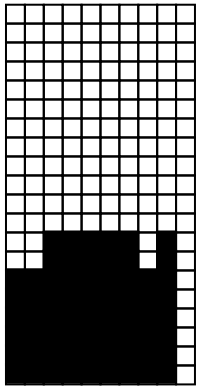
Tetris



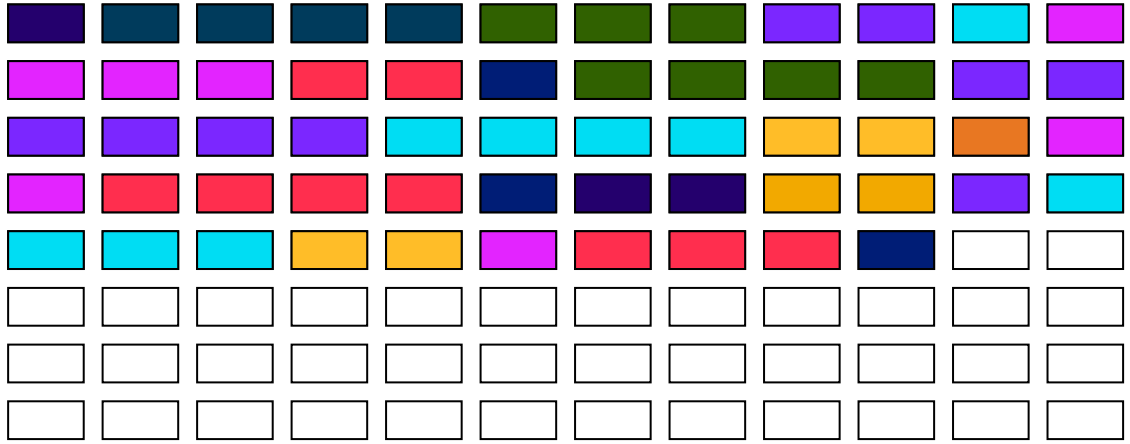


WAFL RAID Parity Calculation

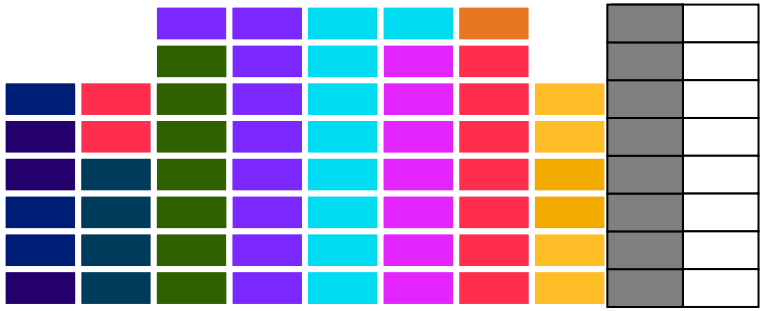
Free Block Bitmap

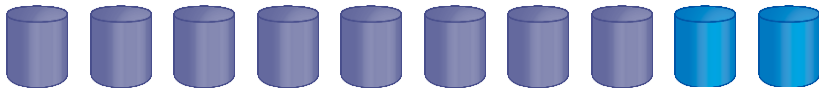


WAFL Cache



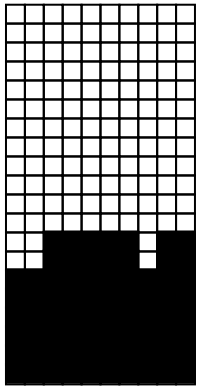
Tetris



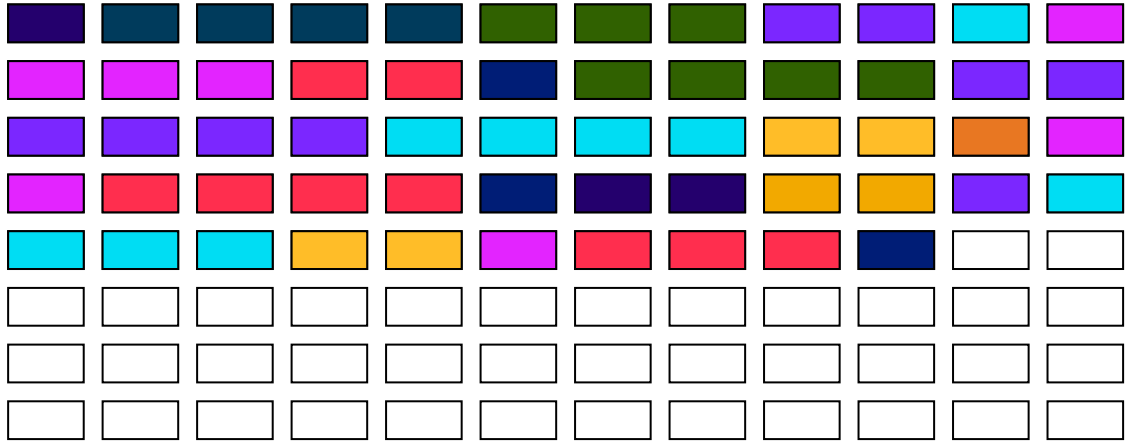


WAFL RAID Diagonal Parity Calculation

Free Block Bitmap

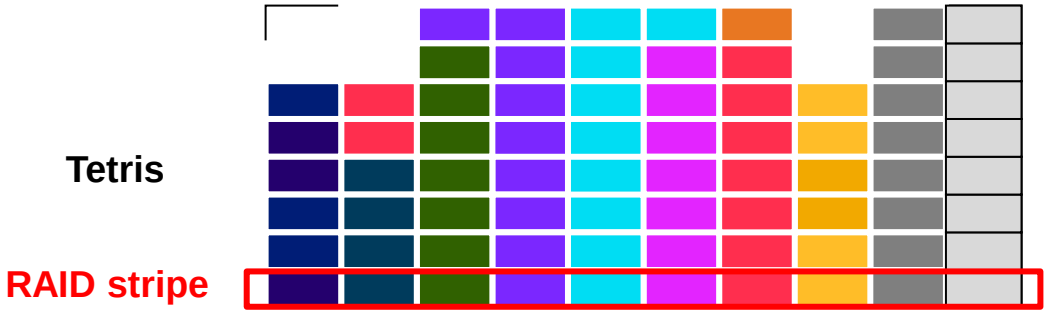


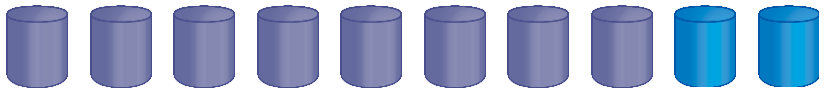
WAFL Cache



Tetris

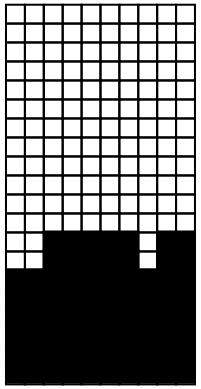
RAID stripe



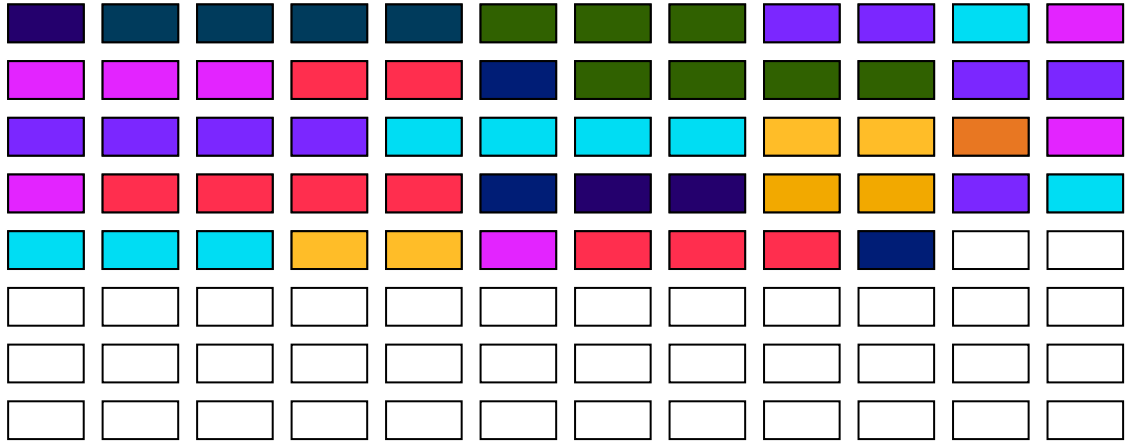


WAFL Disk Write

Free Block Bitmap

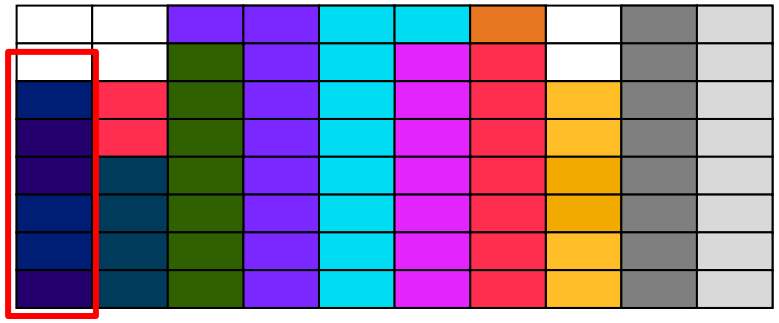


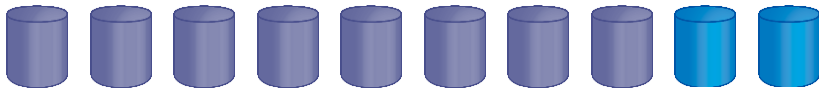
WAFL Cache



Write Chain

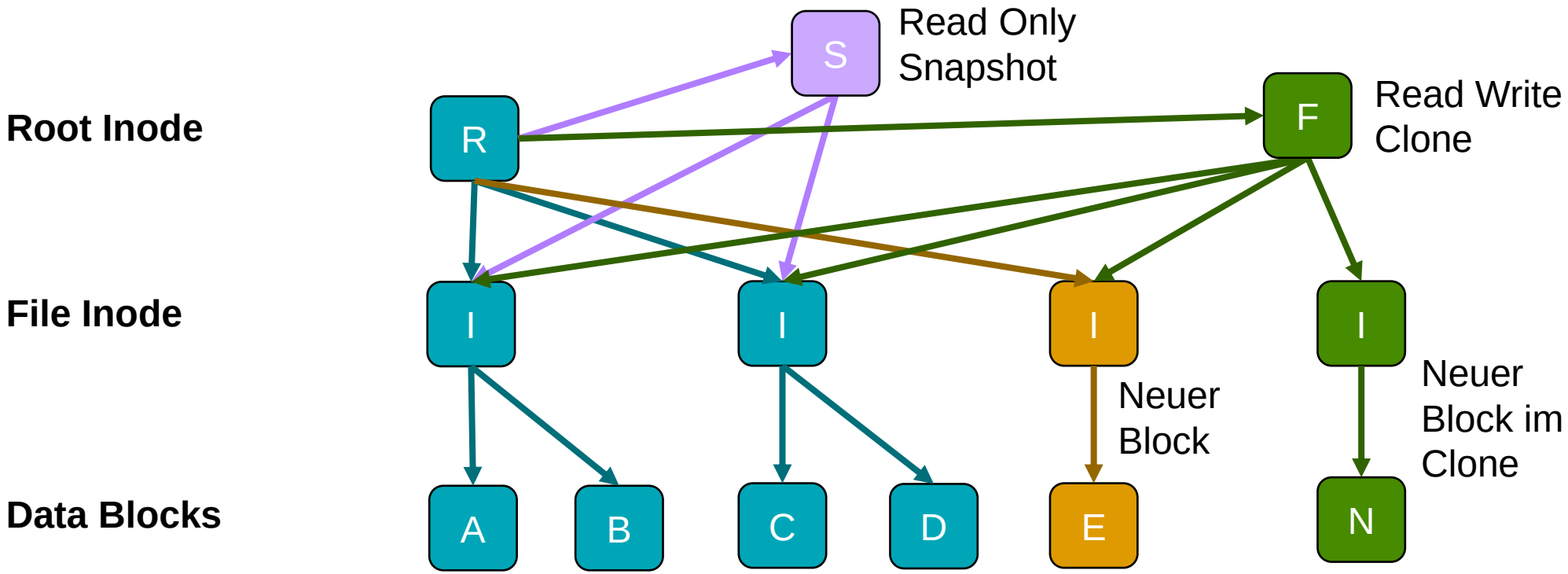
Tetris





ONTAP WAFL

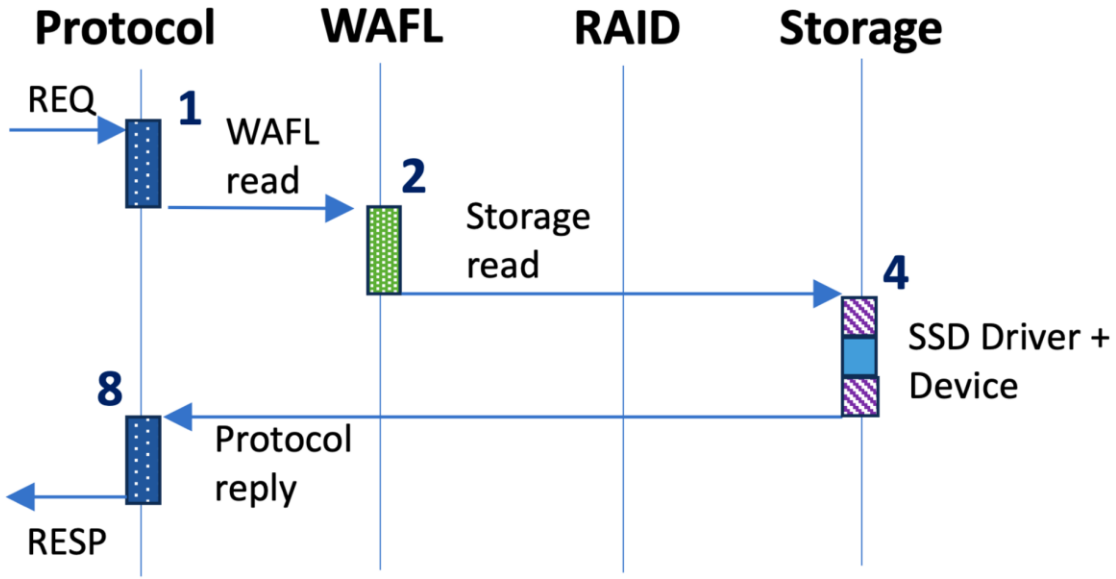
Wie Snapshots und Clones funktionieren



Fastpaths: WAFL Reply and RAID

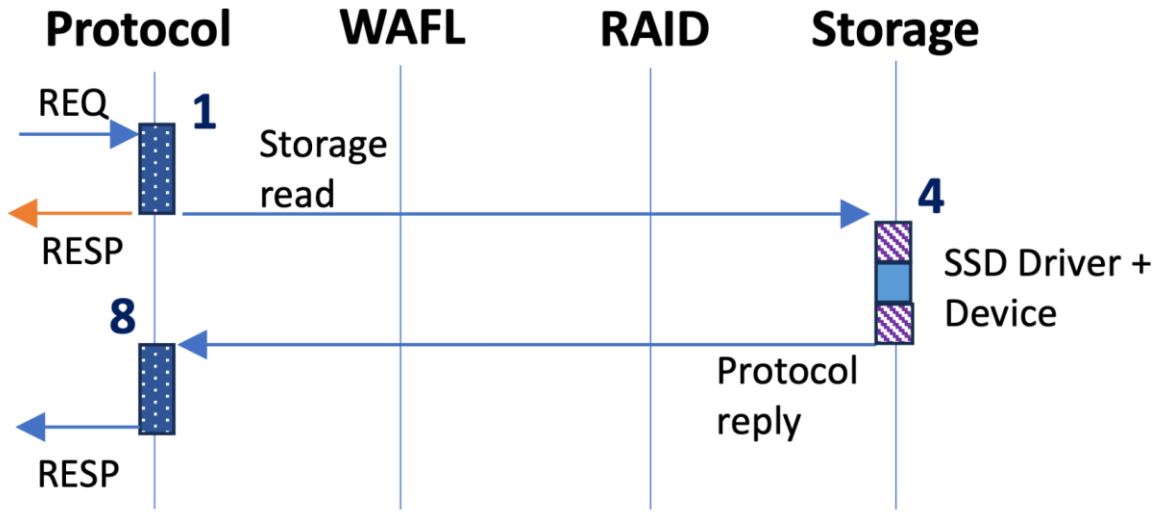
RAID Fastpath

- RAID maintains up-to-date topology of aggregate
  - Drive failures or reconstructs?
  - Mapping PVBN to LBA
- RAID events are rare
  - Exporting read-only cache to WAFL, effectively bypassing RAID
- RAID events will mark the cache stale
  - Storage will detect it in step 4
  - Fall back to legacy read path
  - Fastpath resumes once a valid topology is available
- Legacy read path is used as a fail-safe on any error



TopSpin

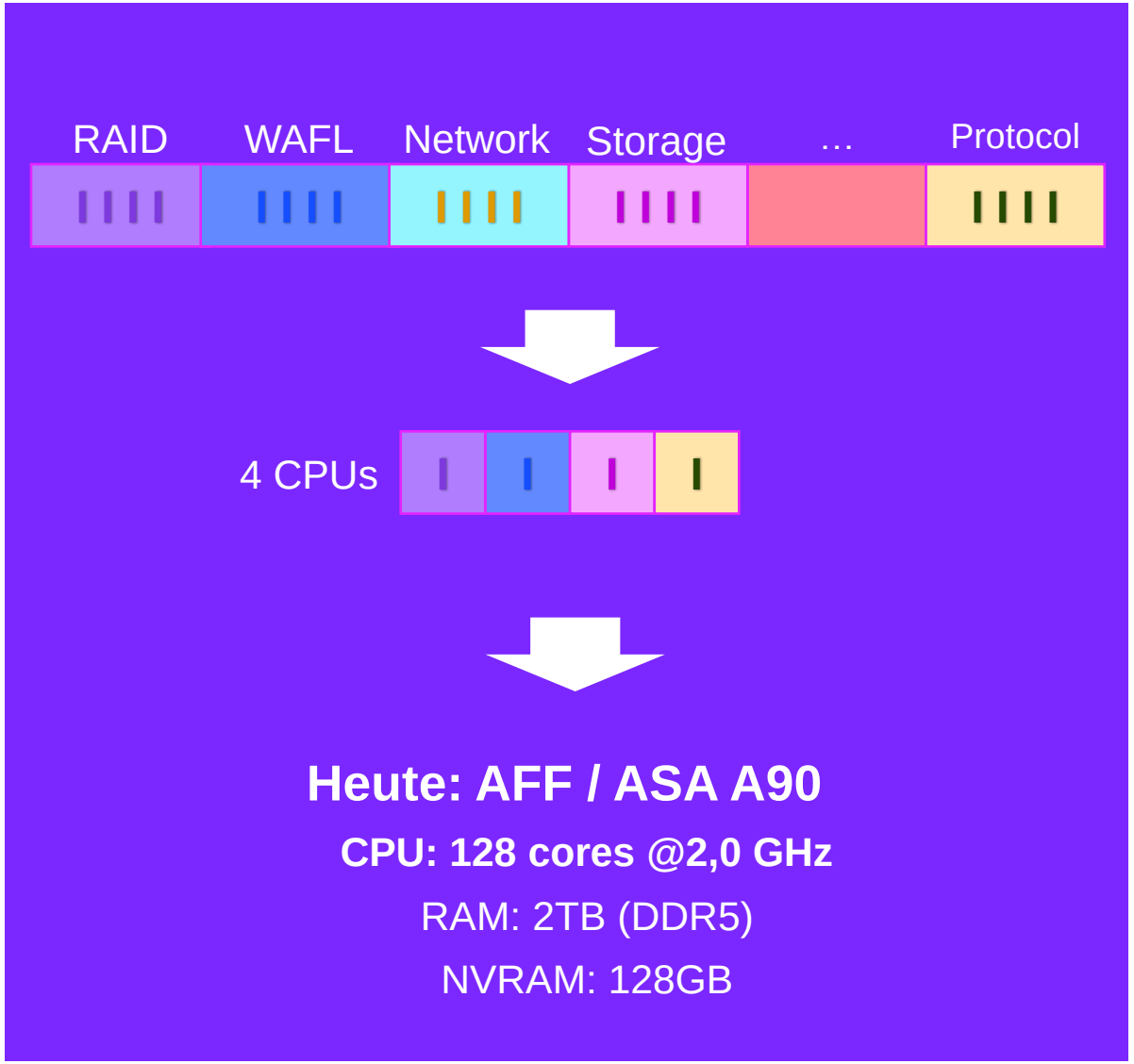
- Protocol has direct access to WAFL data structures
- Protocol component can check if data is in memory or on media
- WAFL component bypassed for reads
- Avoids queueing delays in WAFL
- Less CPU overhead
- Productized in ONTAP 9.3 (2017) for all block-based protocols



Parallelism in ONTAP

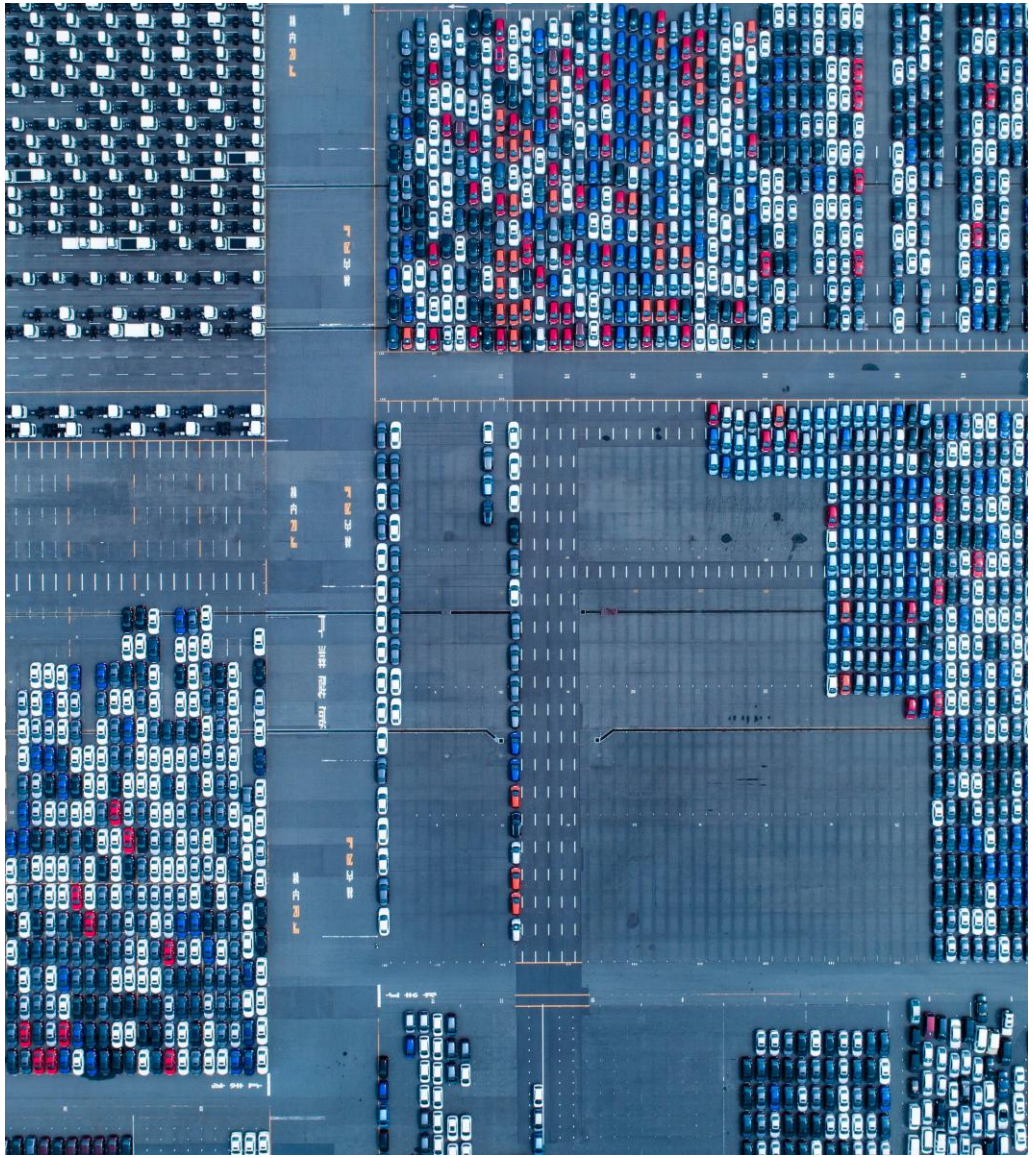
Und warum es heute sehr wichtig ist darauf zu achten

- Data ONTAP was created for single-CPU systems of 1994
- Parallelism via coarse-grained Symmetric Multi-Processing
  - Each subsystem was assigned to a single-threaded domain
  - Minimal explicit locking required
  - Message passing between domains
  - Scaled to 4 cores – but all of WAFL was serialized



From Waffinity to Dynamic Bento (2024)

- **Automated detection of single volume workloads**
- **Dynamic rebalance to reserved Volume affinity**
  - And back to regular Volume affinity
- **Reserved Volume affinity has 21 stripe affinities**
  - Other affinities have 9



CLI to see if Dynamic Bento has been active:

```
node*> wafl volremaphist
```

Optimization for low latency media

Und es hat auch mit SSDs funktioniert

WAFL was designed for hard drives

Block allocator uses multiples of SSD erase page size

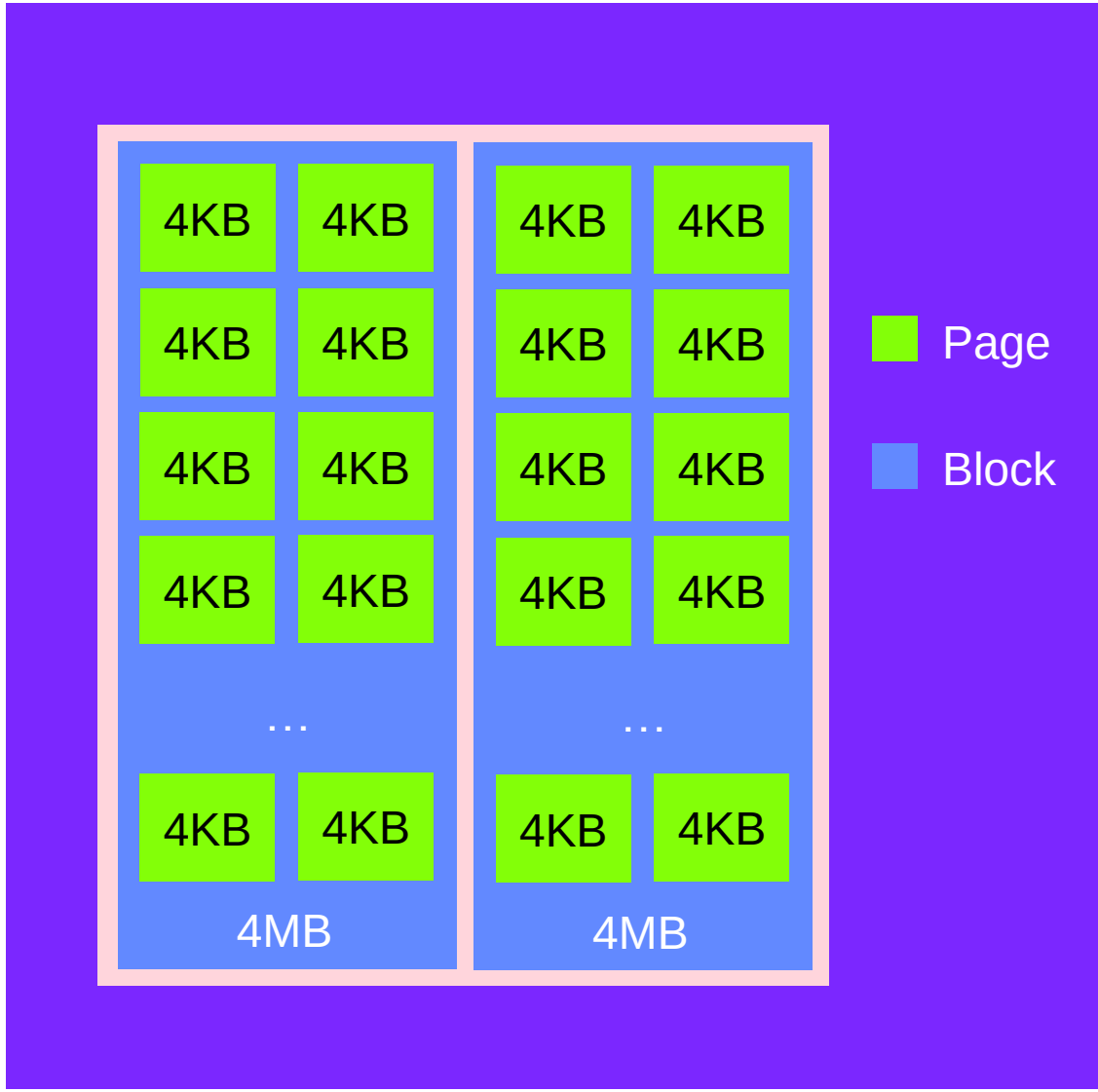
- Mitigating log-on-log problem
- Increasing SSD lifetime

Compression and Deduplication run inline

- Reducing the overall amount of data written to media
- Increasing SSD lifetime

Optimizations in the read path

- Reducing overhead from milliseconds to less than 160us



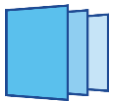
ONTAP Features

Efficiency and Protection

NetApp® Snapshot

Instant recovery points

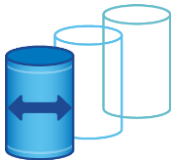
Immutable data copies



Immutable and Indelible

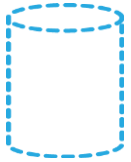
FlexClone® volumes

Writable snapshots, zero capacity penalty, only store writes



Thin provisioning

Grow as you go, and don't over-allocate upfront



Deduplication

Eliminating duplicate blocks found within the disk pool

Inline zero deduplication

Inline & background deduplication

Compression

Shrinking data to reduce overall capacity requirements

Inline & background compression

Hot data compression

Cold data compression

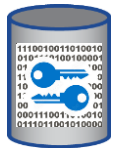
Compaction

Fit more data into a 4 KB block before it's written

Data compaction

Encryption

NetApp Volume Encryption (NVE)
NetApp Aggregate Encryption (NAE)
NetApp Storage Encryption (NSE)



Autonomous Ransomware Protection

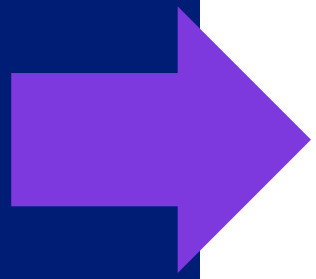
ARP check incoming data and detects Entropy, File extension types, File IOPS

connect TECHNOLOGIE FORUM

ONTAP READ & WRITE WORKFLOW

UND WAS BEDEUTET DAS JETZT?

- WRITES werden im Cache (RAM) verarbeitet und im NVRAM geloggt
- DISKs / SSDs befinden sich nicht im Schreibpfad
- WRITES werden im Cache (RAM) bearbeitet, sortiert und als einheitliche Operation (Stripe) auf die DISKs / SSDs geschrieben
- Stripes / Blöcke werden nicht überschrieben



1. Performance und Einsparungen maximieren
2. Hardware optimal ausnutzen (z.B. SSD Wear vermeiden, gilt auch für HDD)
3. Daten absolut sicher ablegen
4. Für alle Plattformen verfügbar

THAT´S ENOUGH ?

NetApp

connect TECHNOLOGIE FORUM