## FairyWREN: A Sustainable Cache for Write-Read-Erase Interfaces

## Sara McAllister

CMU #UNC Chapel Hill \*Microsoft

**Carnegie Mellon** Parallel Data Laboratory

Sherry (Yucong) Wang, Benjamin Berg<sup>#</sup>, Daniel S. Berger<sup>\*</sup>, George Amvrosiadis, Nathan Beckmann, Gregory R. Ganger

**OSDI 2024** 





## Datacenter emissions need to be curtailed

### Datacenters are projected to emit >33% global emissions by 2050

**Carnegie Mellon Parallel Data Laboratory** 

http://www.pdl.cmu.edu/



ACM TechBrief - Computing and Climate Change '21

#### 40% of server emissions are storage

Lyu HotCarbon '23





# FairyWREN: Enabling Sustainable Caching

- 1) Find that current flash caches have unnecessarily high emissions:
  - Cannot leverage longer lifetimes on denser flash
  - Or rely on too much DRAM
- 2) Describe Write-Read-Erase iNterfaces enable controlling all writes
- 3) FairyWREN leverages WREN to lowers writes by nest packing
- 4) Reduce writes 12.5x over prior flash caches
- 5) Enables low-DRAM, long lifetime, dense flash caching to achieve:



33% reduction in flash emissions





# Flash cache emissions are mostly embodied

#### **Operational Emissions**

Emissions from running the datacenter



#### **Carnegie Mellon** Parallel Data Laboratory

http://www.pdl.cmu.edu/

#### **Embodied Emissions**

Emissions from manufacturing, transportation, raw materials, HW disposal











## Flash cache emissions are mostly embodied

#### **Embodied Emissions**

Emissions from manufacturing, transportation, raw materials, HW disposal



#### 61% of datacenter embodied emissions are storage

#### **Carnegie Mellon**

Parallel Data Laboratory

http://www.pdl.cmu.edu/



### Projected to be 82% of emissions

#### Chasing Carbon - Gupta HPCA 2021

GreenSKU - Wang ISCA '24





## Flash caching: Need low DRAM overhead



**Carnegie Mellon** Parallel Data Laboratory



## Flash caching: Need low DRAM overhead



### 30 bits / object metadata overhead

Flashield (Eisenman NSDI '19)

**Carnegie Mellon** Parallel Data Laboratory

http://www.pdl.cmu.edu/

### 2 TB flash cache → 75 GB memory overhead



## Flash caching: Need low DRAM overhead



#### 30 bits / object metadata overhead

Flashield (Eisenman NSDI '19)

#### Sustainable flash cache constraints: 1) Need a low DRAM overhead solution

**Carnegie Mellon Parallel Data Laboratory** 

http://www.pdl.cmu.edu/

### 2 TB flash cache → 75 GB memory overhead





### Flash has limited write endurance before wearout

**Carnegie Mellon** Parallel Data Laboratory

http://www.pdl.cmu.edu/



### Write Rate = \_\_\_\_\_ Capacity X Write Endurance **Desired Lifetime**



# Caching on flash → Write constraint

### Flash has limited write endurance before wearout

- Capacity X Write Endurance Write Rate = **Desired Lifetime** 2 TB Write Rate = ★ 3000 writes/cell 3 years

**Carnegie Mellon Parallel Data Laboratory** 



# Caching on flash $\rightarrow$ Write constraint

### Flash has limited write endurance before wearout

- **Desired Lifetime** 2 TB Write Rate = ------★ 3000 writes/cell
- 3 years
  - ≈ 63 MB/s

## Strategies to lower emissions will lower write rate

**Carnegie Mellon** Parallel Data Laboratory



# Longer Lifetimes — More Write Constrained

### Longer device lifetimes amortize embodied emissions



**Carnegie Mellon** Parallel Data Laboratory





# Longer Lifetimes — More Write Constrained

### Longer device lifetimes amortize embodied emissions



**Carnegie Mellon Parallel Data Laboratory** 





# Denser Flash — More Write Constrained

### Less HW lowers embodied emissions



**Carnegie Mellon Parallel Data Laboratory** 

https://blocksandfiles.com/2019/08/07/penta-level-cell-flash/, WOM-v Codes (Jaffer FAST '22 )



# Denser Flash — More Write Constrained

### Less HW lowers embodied emissions



**Carnegie Mellon Parallel Data Laboratory** 

https://blocksandfiles.com/2019/08/07/penta-level-cell-flash/, WOM-v Codes (Jaffer FAST '22)







### Less HW lowers embodied emissions



#### Sustainable flash cache constraints: 1) Need a low DRAM overhead solution 2) Need low writes to achieve extended lifetime on dense flash

#### **Carnegie Mellon Parallel Data Laboratory**

https://blocksandfiles.com/2019/08/07/penta-level-cell-flash/, WOM-v Codes (Jaffer FAST '22)

http://www.pdl.cmu.edu/

# **Denser Flash** $\rightarrow$ More Write Constrained



# Logical-Block-Addressable Devices (LBAD)



**Carnegie Mellon** Parallel Data Laboratory









**Carnegie Mellon** Parallel Data Laboratory

http://www.pdl.cmu.edu/

## LBAD devices require device GC

#### Low DRAM caches use hashing creating random writes







**Carnegie Mellon** Parallel Data Laboratory

http://www.pdl.cmu.edu/

## LBAD devices require device GC

#### Low DRAM caches use hashing creating random writes



## LBAD devices require device GC



**Carnegie Mellon** Parallel Data Laboratory





## LBAD devices require device GC



## LBAD devices require device GC



# Write-Read-Erase iNterfaces (WREN)



Erase Unit

**Carnegie Mellon** Parallel Data Laboratory



# Write-Read-Erase iNterfaces (WREN)



**Carnegie Mellon** Parallel Data Laboratory





# Write-Read-Erase iNterfaces (WREN)



### WREN Interfaces (ZNS, FDP) allow caches to control all writes

**Carnegie Mellon** 

Parallel Data Laboratory





# FairyWREN: Leveraging control over erases



**Carnegie Mellon** 

Parallel Data Laboratory

http://www.pdl.cmu.edu/

### Small object cache is main source of writes, based on Kangaroo

McAllister SOSP '21





# FairyWREN: Leveraging control over erases



**Carnegie Mellon** 

Parallel Data Laboratory

http://www.pdl.cmu.edu/

### Small object cache is main source of writes, based on Kangaroo

McAllister SOSP '21





#### Cache Logic

Victim Segment

#### Garbage Collection

Victim EU ...

**Carnegie Mellon** Parallel Data Laboratory





**Carnegie Mellon** Parallel Data Laboratory







**Carnegie Mellon** Parallel Data Laboratory





### Nest packing reduces writes by 3.7x

**Carnegie Mellon** Parallel Data Laboratory



# Other contributions in FairyWREN



#### **Carnegie Mellon** Parallel Data Laboratory

- Write improvement of large-small object separation
- Hot-cold separation within sets
  - 3.4x write reduction
- Sliced log design leveraging double buffering 8.3 bits/object memory overhead





# FairyWREN cuts writes by 12.5x



#### **Carnegie Mellon** Parallel Data Laboratory

1.5	2.0	2.5
Days		



# FairyWREN cuts writes by 12.5x



**Carnegie Mellon** Parallel Data Laboratory





**Carnegie Mellon Parallel Data Laboratory** 

http://www.pdl.cmu.edu/

Caching server emissions for 6 year lifetime, 30% miss ratio target on a Twitter trace







**Carnegie Mellon Parallel Data Laboratory** 

http://www.pdl.cmu.edu/

#### Caching server emissions for 6 year lifetime, 30% miss ratio target on a Twitter trace

Minimum Writes (+): No write amplification, no DRAM overhead







#### **Carnegie Mellon**

**Parallel Data Laboratory** 

http://www.pdl.cmu.edu/

Caching server emissions for 6 year lifetime, 30% miss ratio target on a Twitter trace

Flashield (V): No write amplification, too much DRAM







#### **Carnegie Mellon**

**Parallel Data Laboratory** 

http://www.pdl.cmu.edu/

Caching server emissions for 6 year lifetime, 30% miss ratio target on a Twitter trace

Kangaroo (M): Too many writes with sustainable flash trends





# FairyWREN: 33% reduction in flash emissions

Caching server emissions for 6 year lifetime, 30% miss ratio target on a Twitter trace



**Carnegie Mellon Parallel Data Laboratory** 

![](_page_38_Picture_8.jpeg)

![](_page_38_Picture_9.jpeg)

# FairyWREN: Enabling Sustainable Caching

- 1) Find that current flash caches have unnecessarily high emissions:
  - Cannot leverage longer lifetimes on denser flash
  - Or rely on too much DRAM
- 2) Describe Write-Read-Erase iNterfaces enable controlling all writes
- 3) FairyWREN leverages WREN to lowers writes by nest packing
- 4) Reduce writes 12.5x over prior flash caches
- 5) Enables low-DRAM, long lifetime, dense flash caching to achieve:

![](_page_39_Picture_11.jpeg)

33% reduction in flash emissions

![](_page_39_Picture_15.jpeg)

![](_page_39_Picture_16.jpeg)