Amnesic **C**ache **M**anagement for Non-Volatile Memory

Dongwoo Kang, Seungjae Baek, Jongmoo Choi
Dankook University, South Korea
{**kangdw,** baeksj, chiojm}@dankook.ac.kr

Donghee Lee
University of Seoul, South Korea
dhl_express@uos.ac.kr

Sam H. Noh
Hongik University, South Korea
samhnoh@hongik.ac.kr

Onur Mutlu
Carnegie Mellon University, USA
onur@cmu.edu

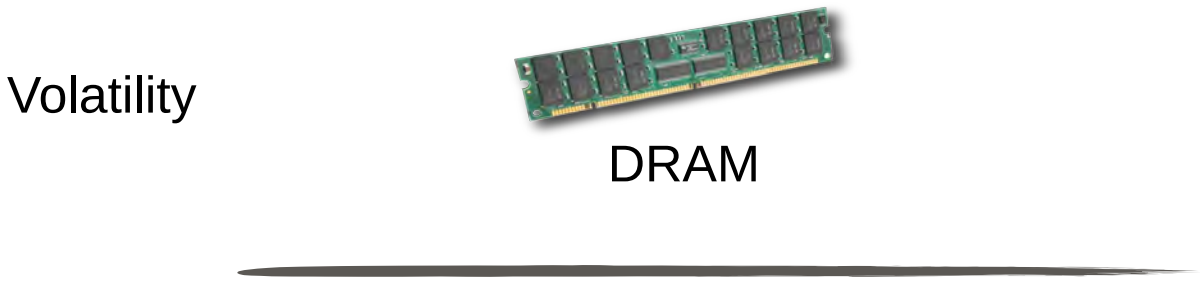
31st International Conference on **M**assive **S**torage **S**ystems and **T**echnology

Outline

- Introduction & Motivation
  - Non-Volatile Memory
  - Phase Change Memory
  - Caching Time
- Design
- Evaluation
- Conclusion

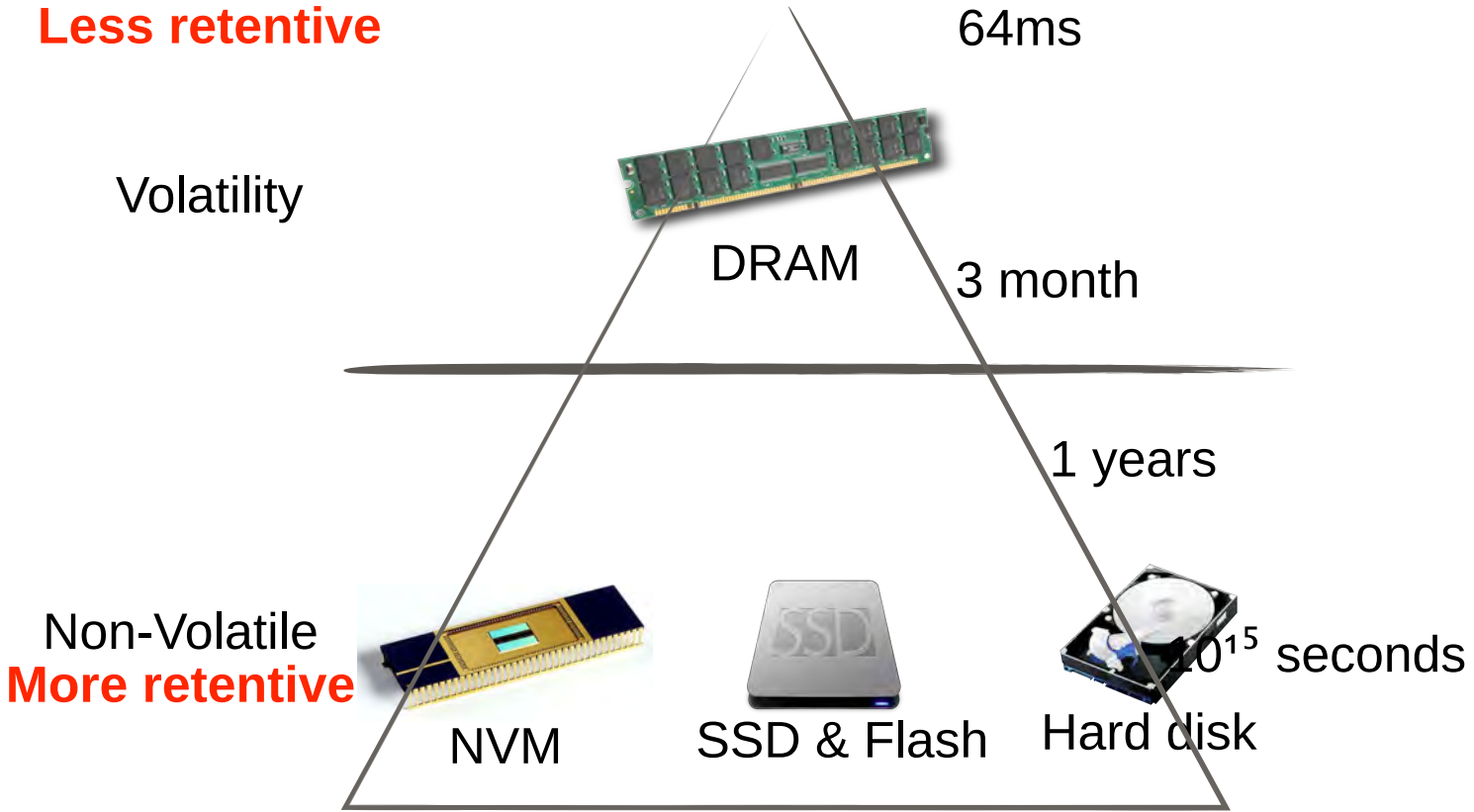
Introduction : Volatility

- **Non-Volatile Memory**
  - PCM (Phase Change Memory), STT-RAM (Spin Transfer Torque RAM), ReRAM (Resistive RAM), Fe-RAM (Ferroelectric Random Access Memory)
  - Byte addressability and Non-Volatility
  - RAM, storage, file cache, CPU cache



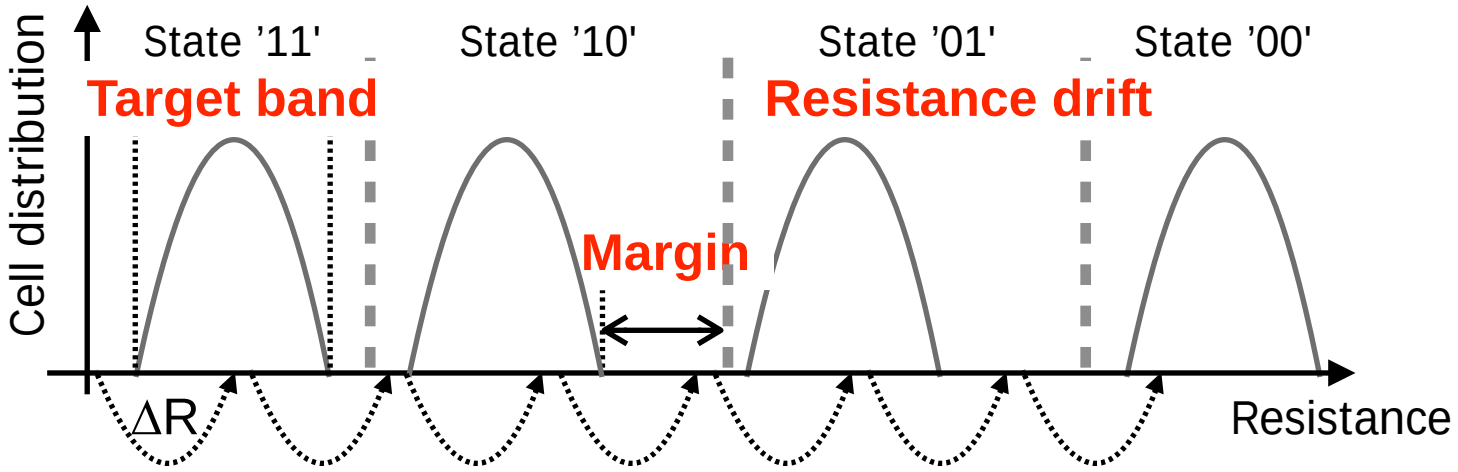
Introduction : Volatility

- **Non-Volatile Memory**
  - PCM (Phase Change Memory), STT-RAM (Spin Transfer Torque RAM), ReRAM (Resistive RAM), Fe-RAM (Ferroelectric Random Access Memory)
  - Byte addressability and Non-Volatility
  - RAM, storage, file cache, CPU cache
  - Limited retention capability, relaxation write



Introduction : Phase Change Memory

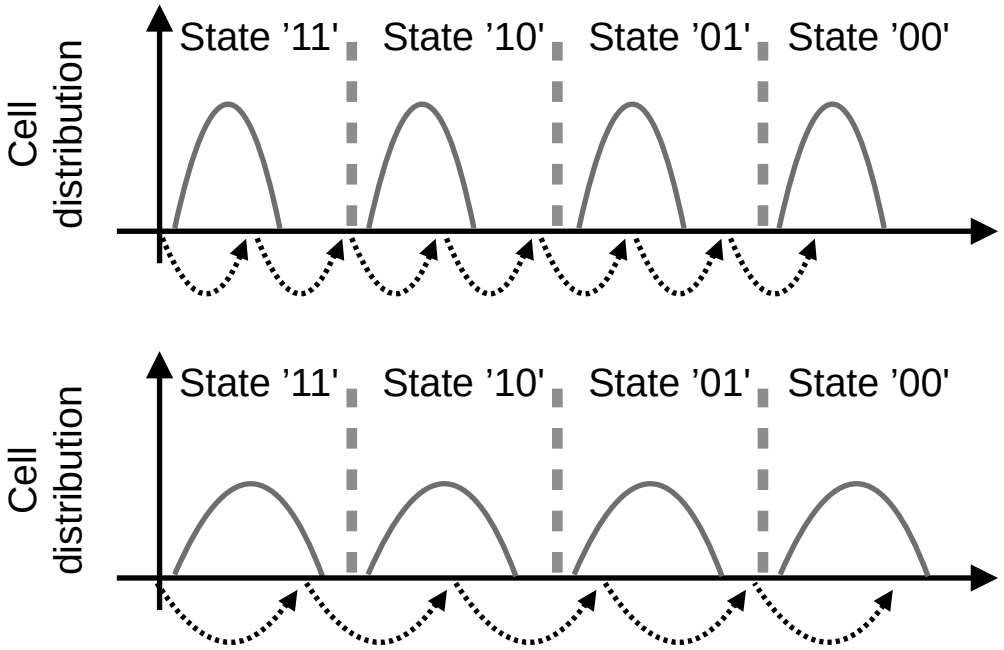
- **States of PCM (Phase Change Memory)**
  - Target band
    - A region of resistances that corresponds to valid bits
  - Write scheme
    - PCM adopts iterative write scheme
    - The resistance of a cell is determined according to the width of the target band.
  - Resistance drifts
    - The resistance in a PCM cell has a tendency to increase by time
    - When the resistance drifts up to the boundary of the next region, the state can be incorrectly represented leading to data loss

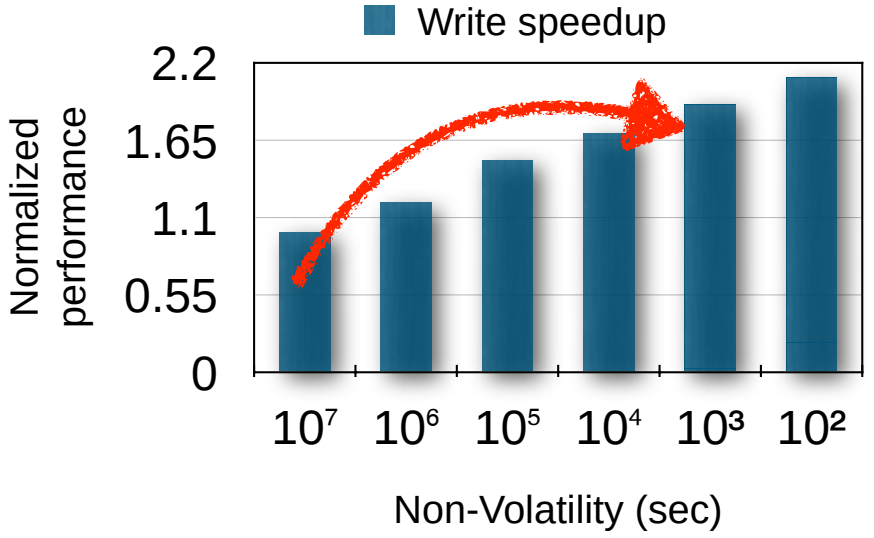


Introduction : Tradeoff

- **Tradeoff between retention capability and write speed**
  - Narrowing target bands
    - Requires more precise control over the iterative mechanism
    - Demands smaller ΔR resulting in a slowdown of the write latency
  - Higher retention increasing write latency
    - 1.7x write speedup can be obtained by reducing the retention capability of PCM from 10^7 to 10^4 seconds [Liu et al.]

How to exploit these characteristics of the PCM?



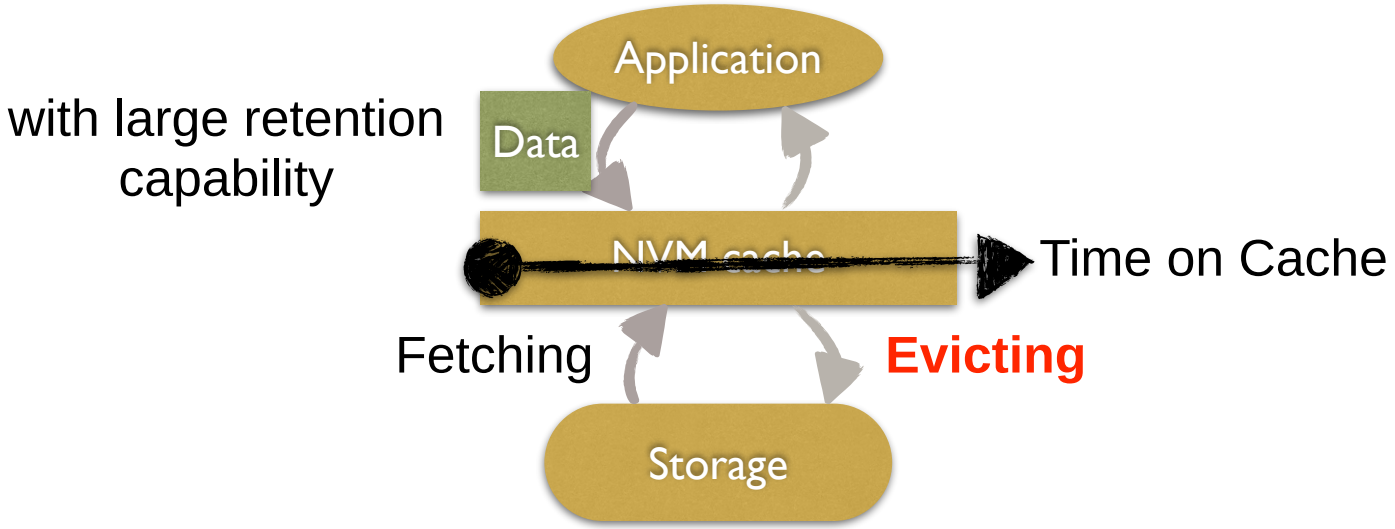


(source :Liu et al., ASPLOS '14)

Motivation : What about NVM cache?

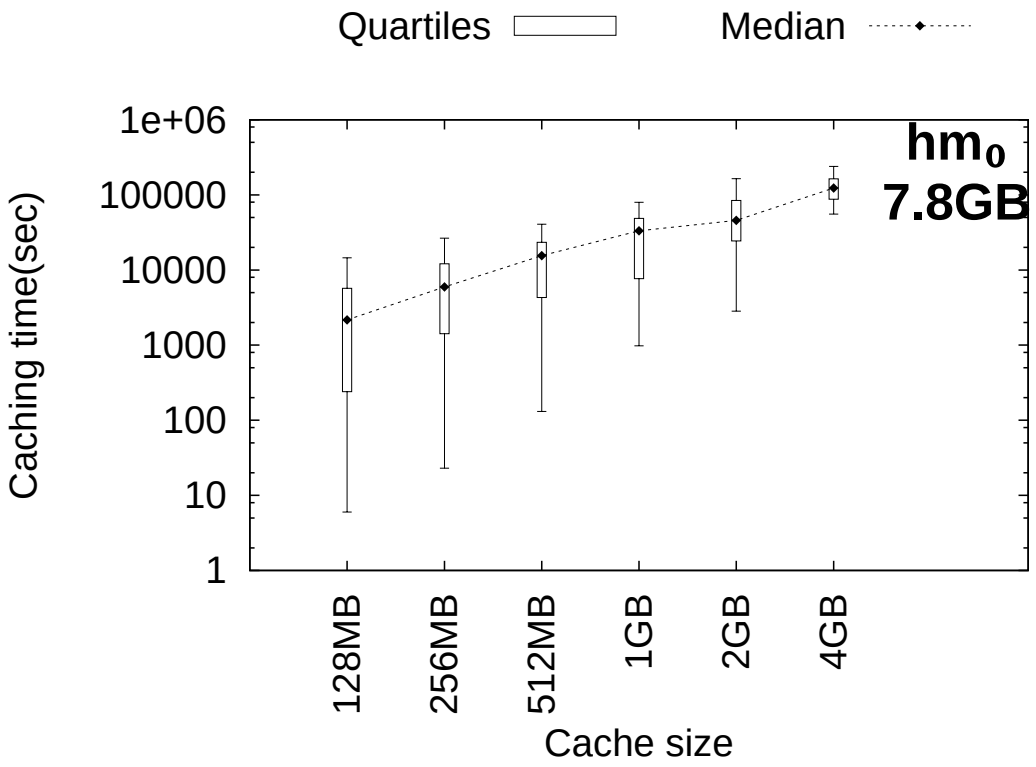
- NVM Cache
  - Employing an NVM cache provides performance improvements
  - Fetching/Eviction data from/to storage system
- Retention capability for the cache
  - 10^7 seconds is recommended retention capability from JEDEC
  - But, data will be evicted from the NVM cache
  - Ensure retention capability while the data is in the cache

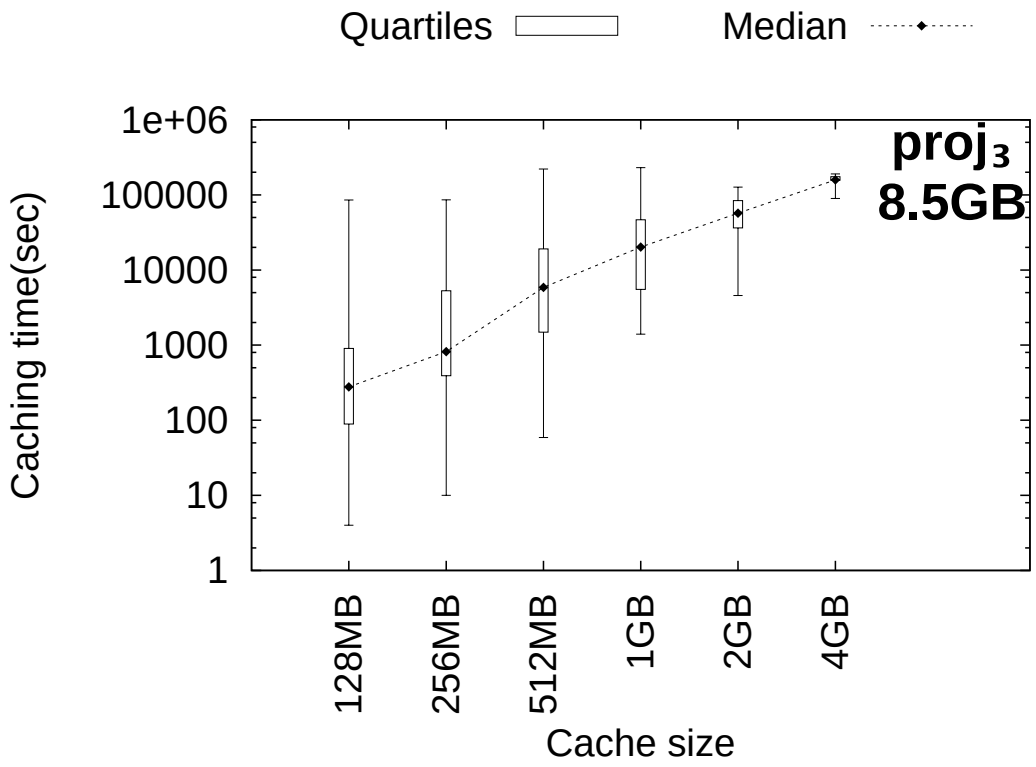
How much retention capability is required with the NVM cache?



Motivation : Caching time

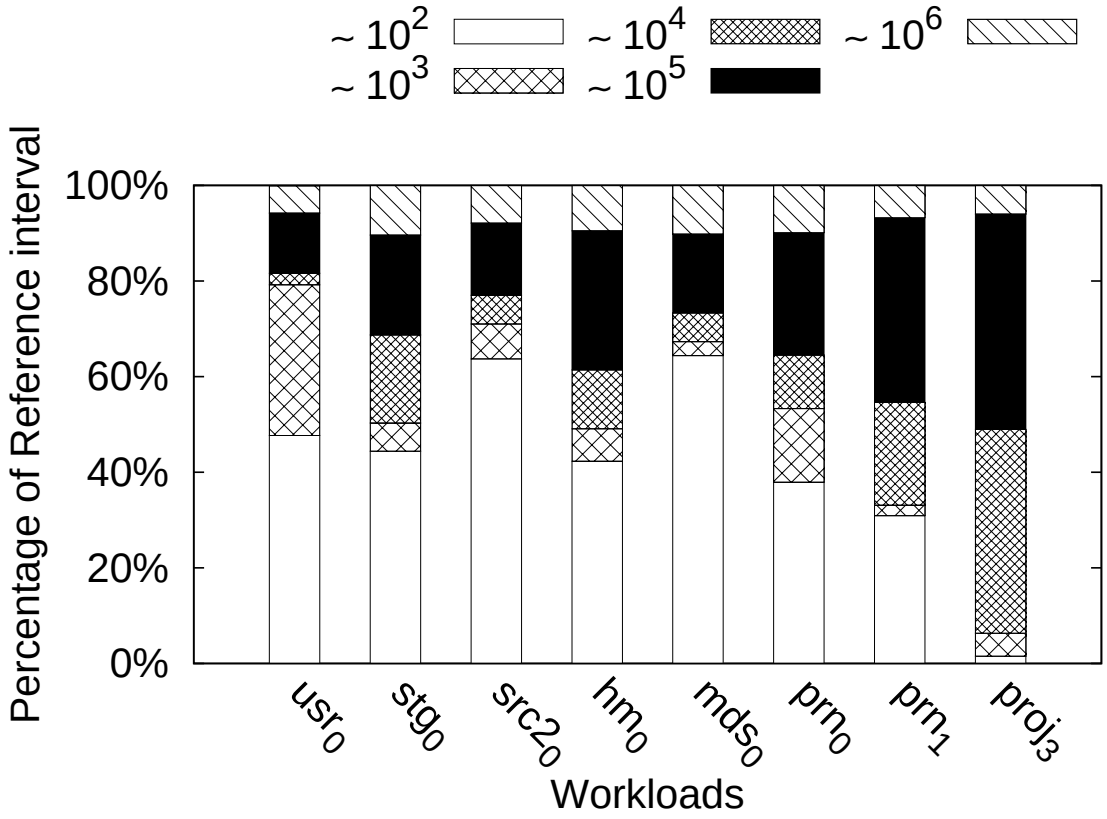
- **Caching time on the NVM cache**
  - We measure the caching time with LRU scheme
  - $T_{Caching} = T_{Evict} - T_{First}$
  - 75% of the data is less than 10^5 seconds
  - Don't need to ensure 10^7 seconds retention capability in the cache





Motivation : Reference interval

- **Reference interval**
  - 90% of data are re-referenced within the 10^5 second interval
  - Retention relaxation can enhance write performance
  - However, when data is re-referenced after its retention capability, it will induce a miss, reducing the hit ratio and triggering extra accesses to retrieve the data from storage.



Motivation : Amnesic technique

Write Performance Improvement

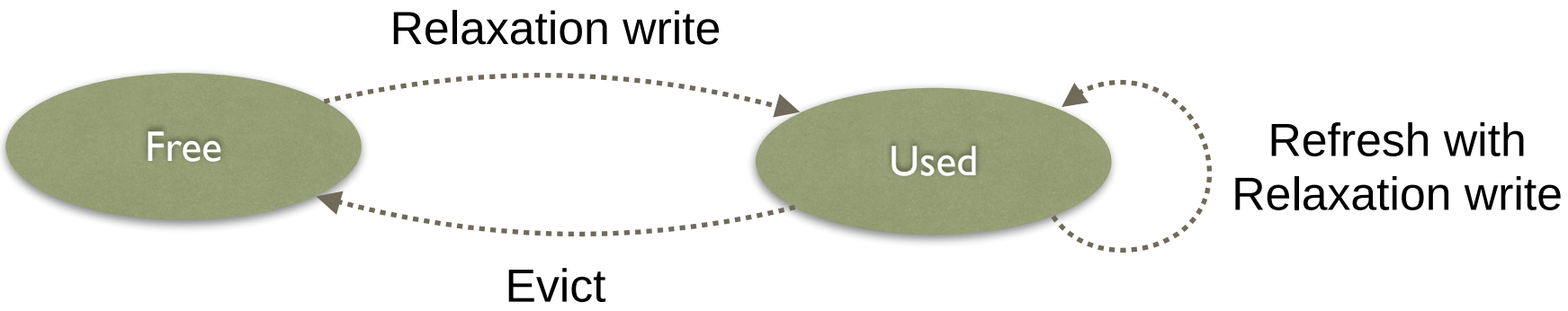
Decreasing Hit Ratio

Outline

- Introduction & Motivation
- Design
  - REF
  - SACM
  - AACM
- Evaluation
- Conclusion

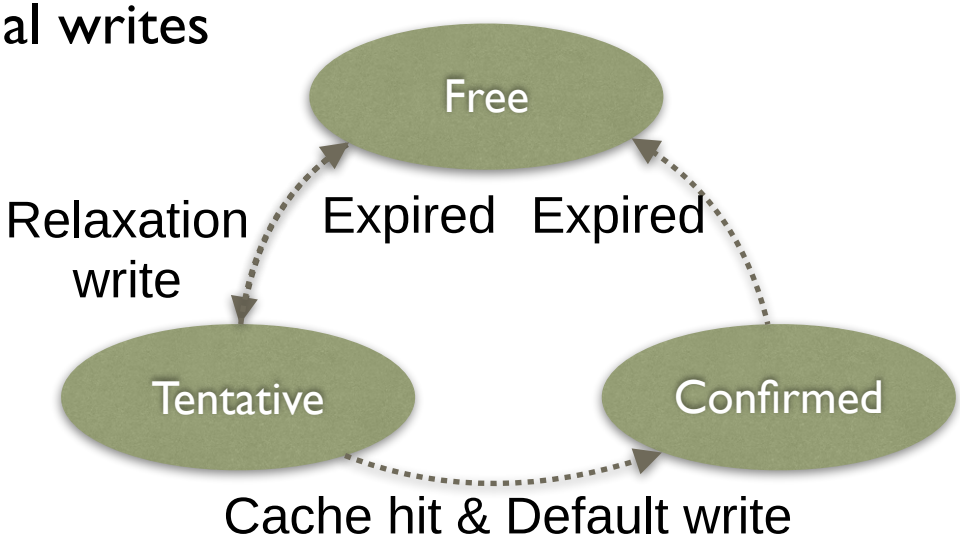
Design : REF

- **REF(REFresh-based cache management scheme)**
  - REF is similar to the LRU scheme
  - Free state and Used state
  - Enhances write speed by relaxing retention capability from 10^7 to 10^4
    - Write latency is decrease by 1.7X
  - Performs refreshing for data whose retention time is about to expire
  - Issue
    - Refresh operation



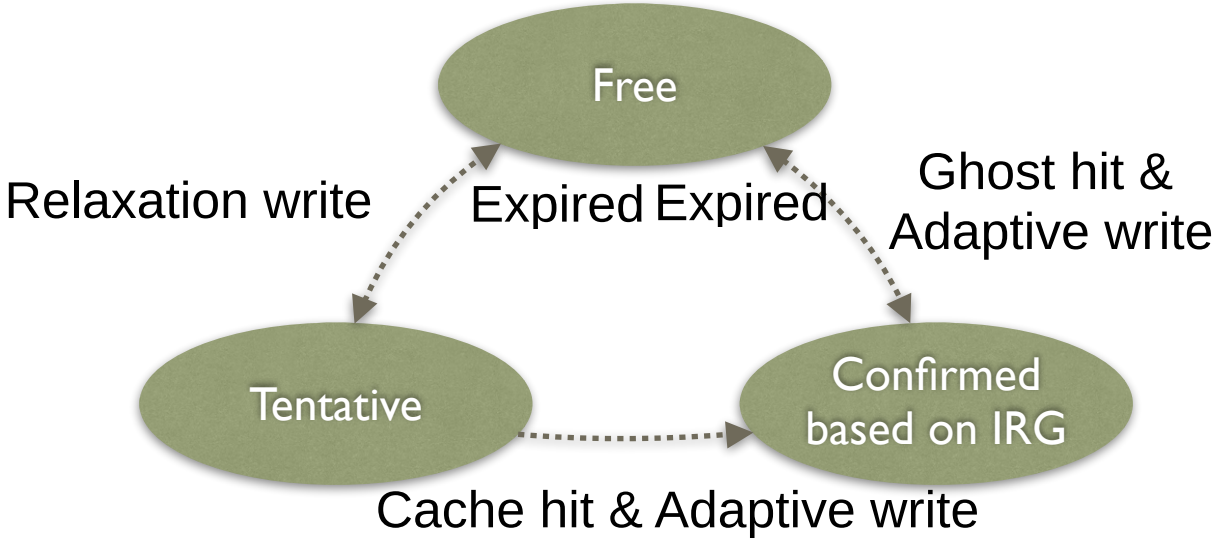
Design : SACM

- **Simple Amnesic Cache Management**
  - Free State to Tentative State
    - Initial write into the cache, the datum is written with the relaxed write(10^4)
  - Tentative State to Confirmed State
    - If it is referenced again within the retention time
    - It is rewritten with 10^7 retention capability
  - Confirmed State to Free State
    - If it is not referenced again and the retention time expires
  - Issue
    - Additional writes



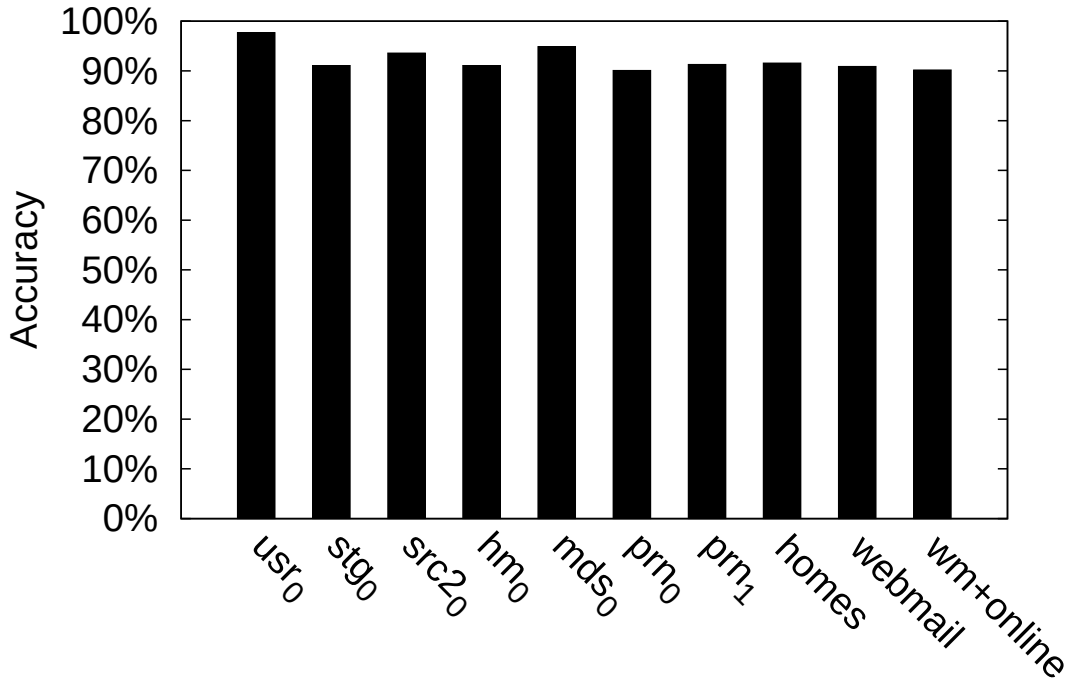
Design : AACM (1/3)

- **Adaptive Amnesic Cache Management**
  - Key idea
    - Estimates the next reference of each data and adaptive write
  - Estimation by IRG model
    - Use 1st order Markov chain for estimation of IRG
  - Adaptive write
    - Ensure appropriate retention capability adaptively for each data
  - Ghost buffer
  - Issue
    - Adaptive write and Estimation



Design : AACM (2/3)

- **Estimation of IRG**
  - Coarse grain levels
    - 10^2, 10^3, 10^4, 10^5, 10^6, 10^7 seconds
  - Accuracy is larger than 90%
  - Memory overhead is 144 bytes for each data
  - Ghost buffer maintains information of 1K blocks.
  - AACM needs the refresh operations for the read request if the remaining retention capability is shorter than the predicted IRG.



Outline

- Introduction & Motivation
- Design
- Evaluation
- Conclusion

Evaluation : Environment

- **Simulator**
  - Time accurate in-house simulator
  - Storage simulator and trace replayer

- **Trace**
  - MSR-Cambridge traces (for 7 days)
  - FIU traces during (for days)
  - Websearch3 trace (for 3.1 days)

- **Simulator parameters**

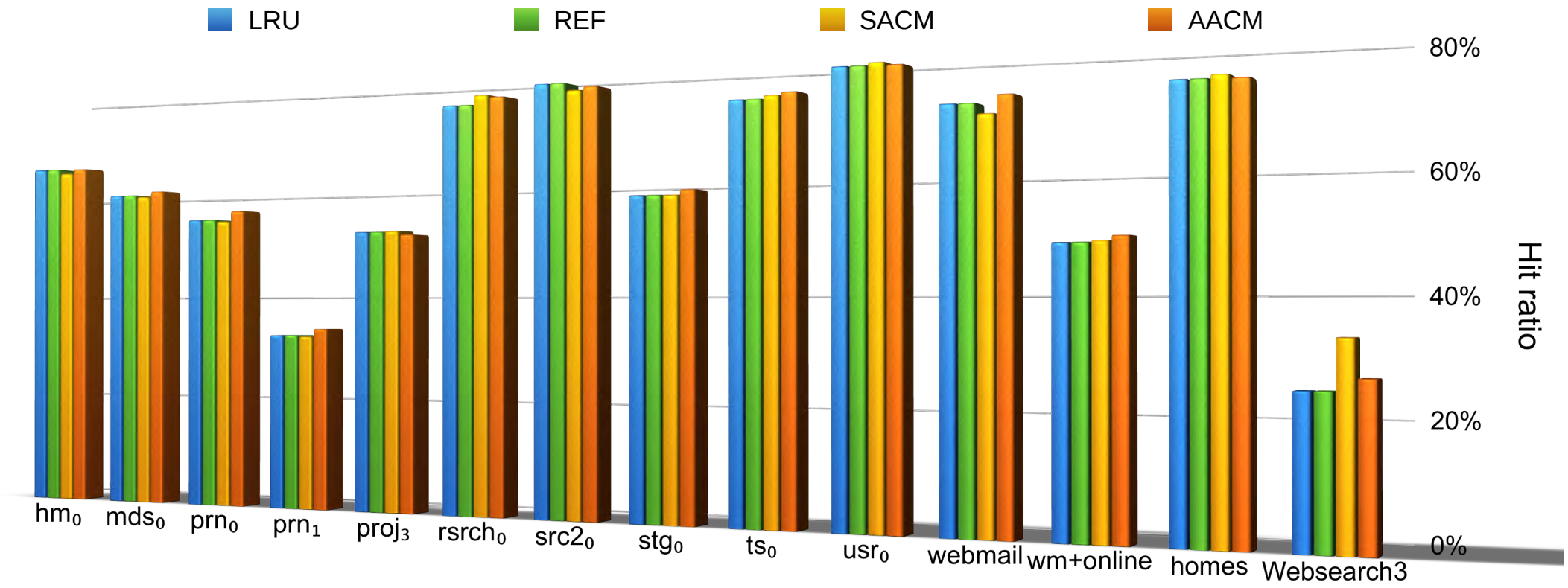
(source :Liu et al., ASPLOS '14)

	PCM	SSD
READ LATENCY	16 us	50 us
WRITE LATENCY	91.2 us	900 us
READ ENERGY	81.9 nj	14.25uj
WRITE ENERGY	4.73 uj	256 uj

RETENTION	SPEEDUP
10^7	1X
10^6	1.2X
10^5	1.5X
10^4	1.7X
10^3	1.9X
10^2	2.1X

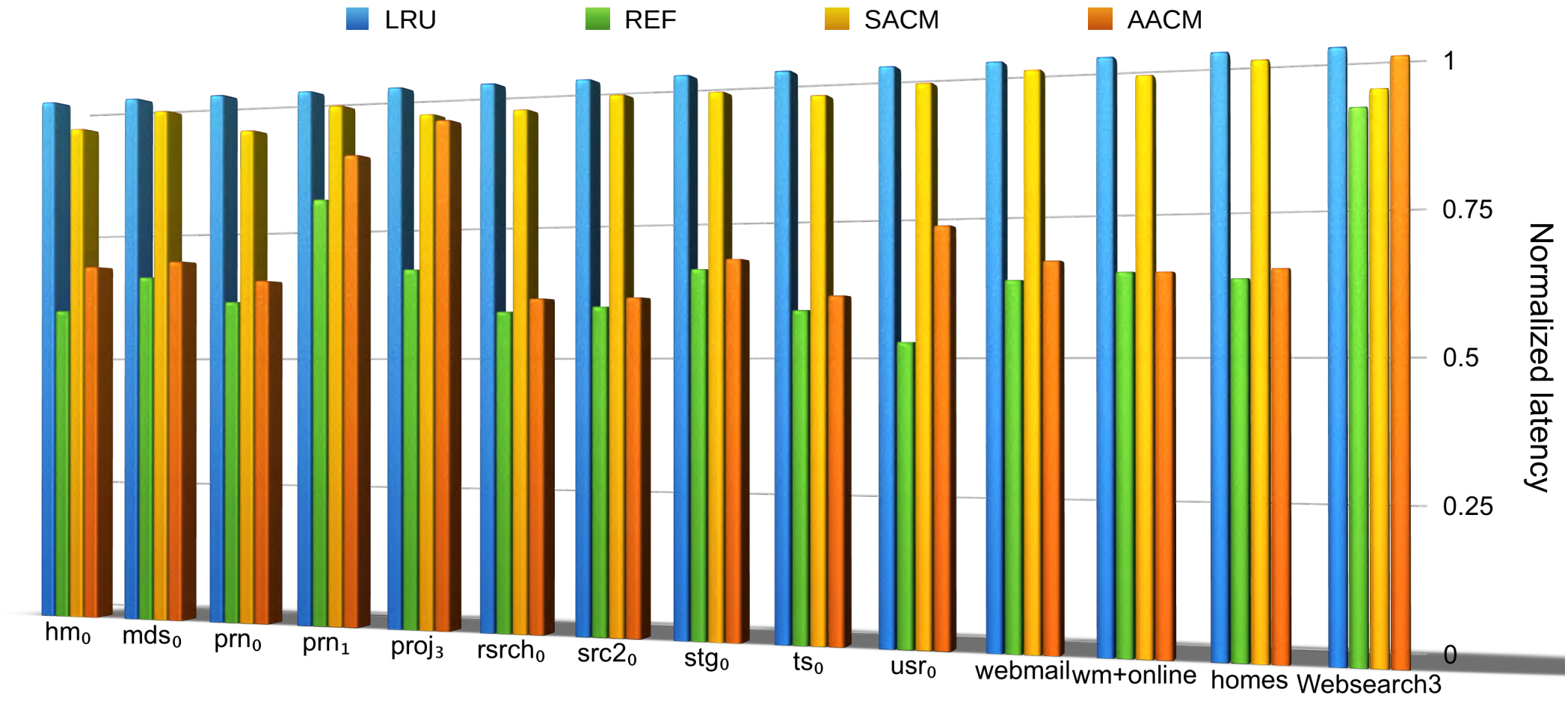
Evaluation : Hit ratio

- **Hit ratio**
  - Cache size is set to 25 % of working set of each workload
    - Cache size is set to be 1.95GB with hm_0 trace(the working set is 7.8GB)
  - Comparable to LRU giving and taking a little bit depending on the workload



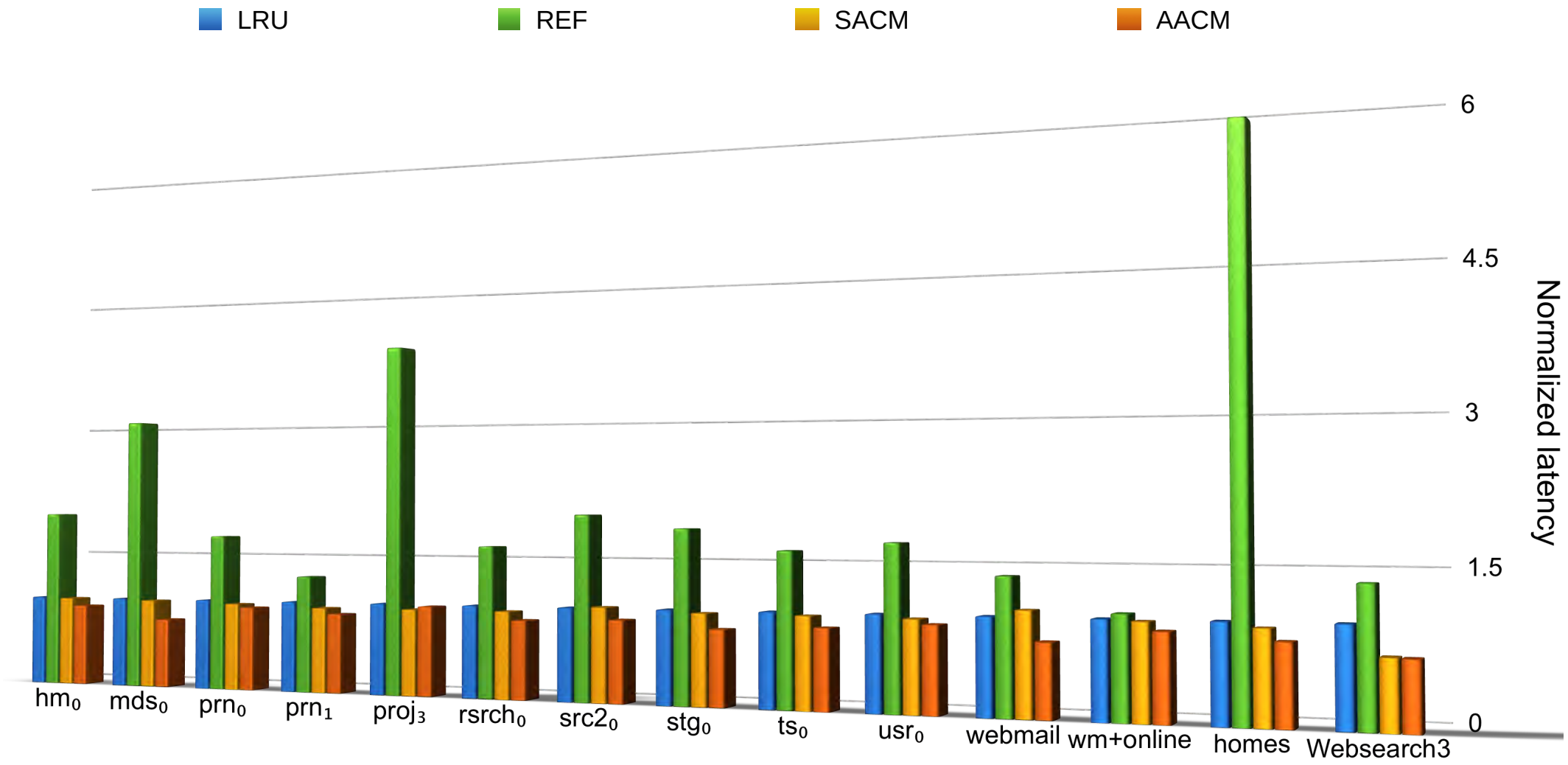
Evaluation : Latency

- **Latency (normalized to LRU)**
  - REF reduces latency even more by as much as 48% (36% on average)
  - SACM does it by as much as 7% (4% on average)
  - AACM does it up to 40% (30% on average)



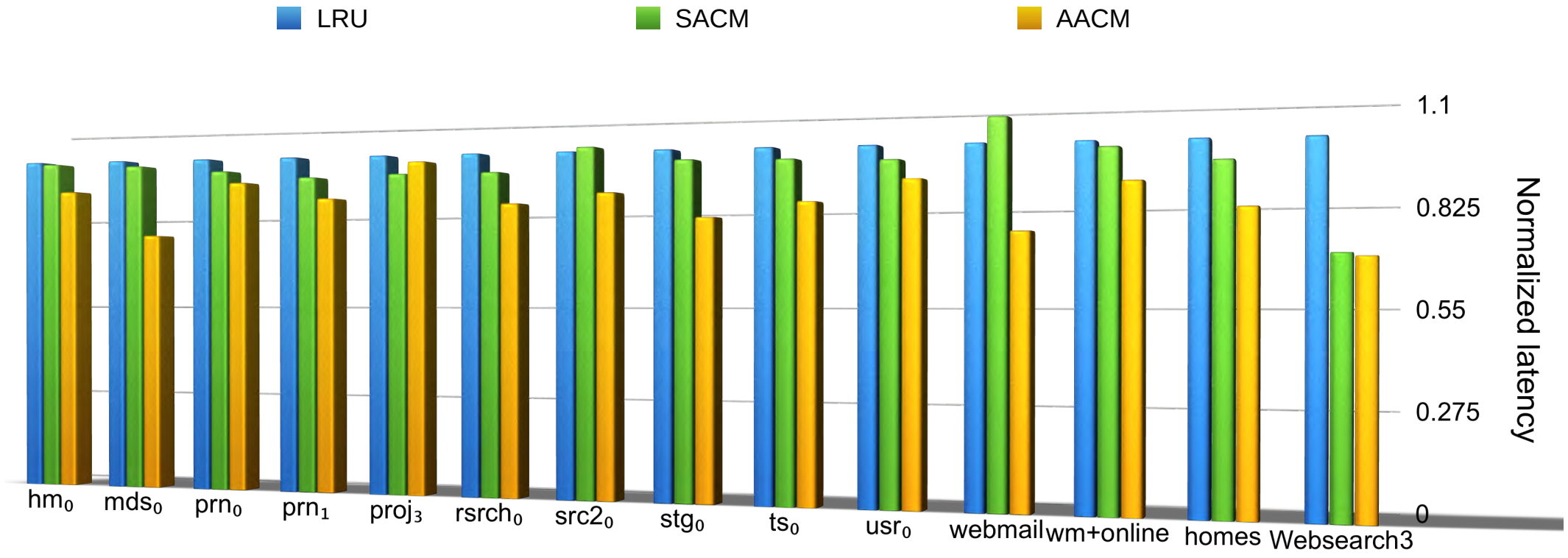
Evaluation : Latency with refresh

- **Latency (normalized to that of LRU)**
  - REF with refresh operations increases normalized latency up to 6X



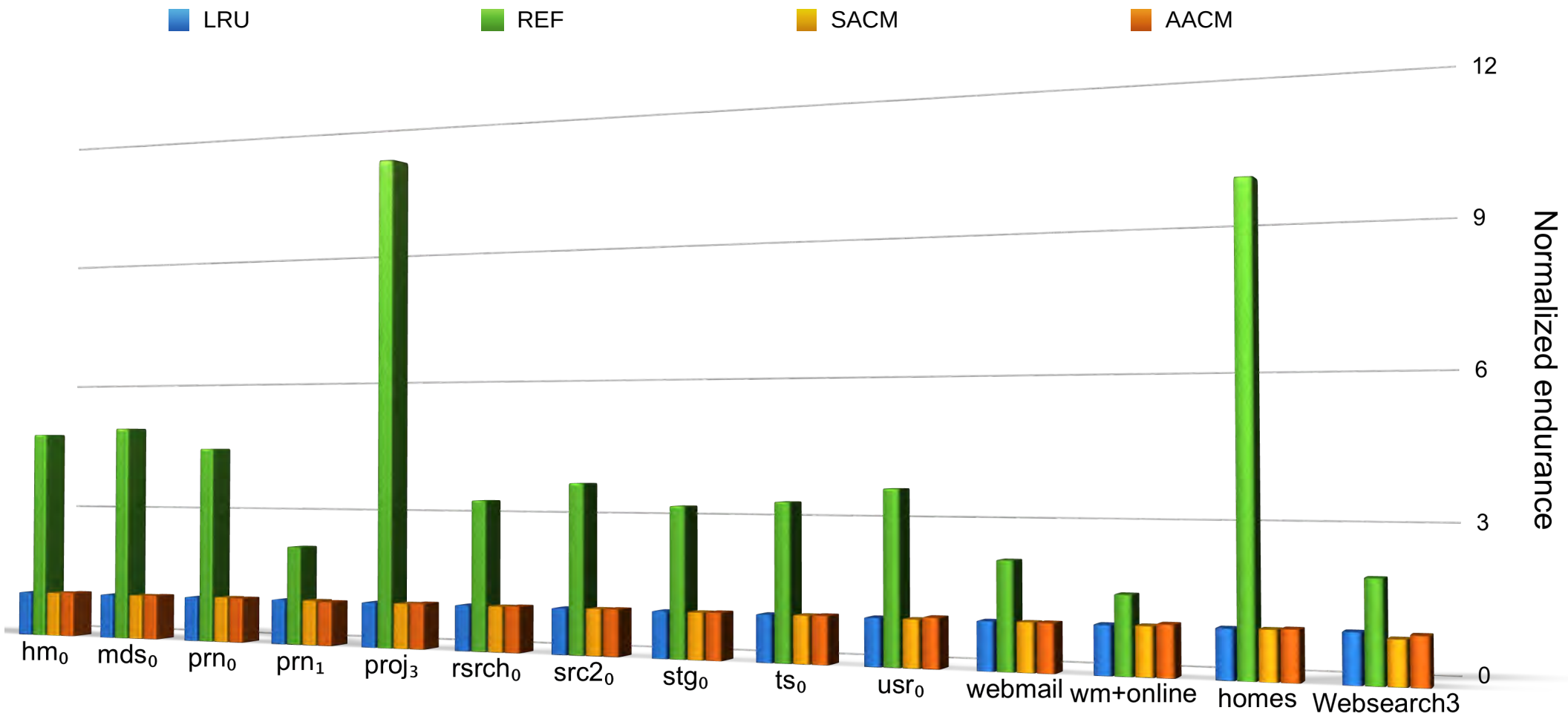
Evaluation : Latency with refresh (without REF)

- **Latency (normalized to that of LRU)**
  - REF with refresh operations increases normalized latency up to 6X
  - SACM and AACM perform better than LRU though the margin has dwindled
    - SACM decreases the latency by 5% on average
    - AACM decreases the latency by 15% on average



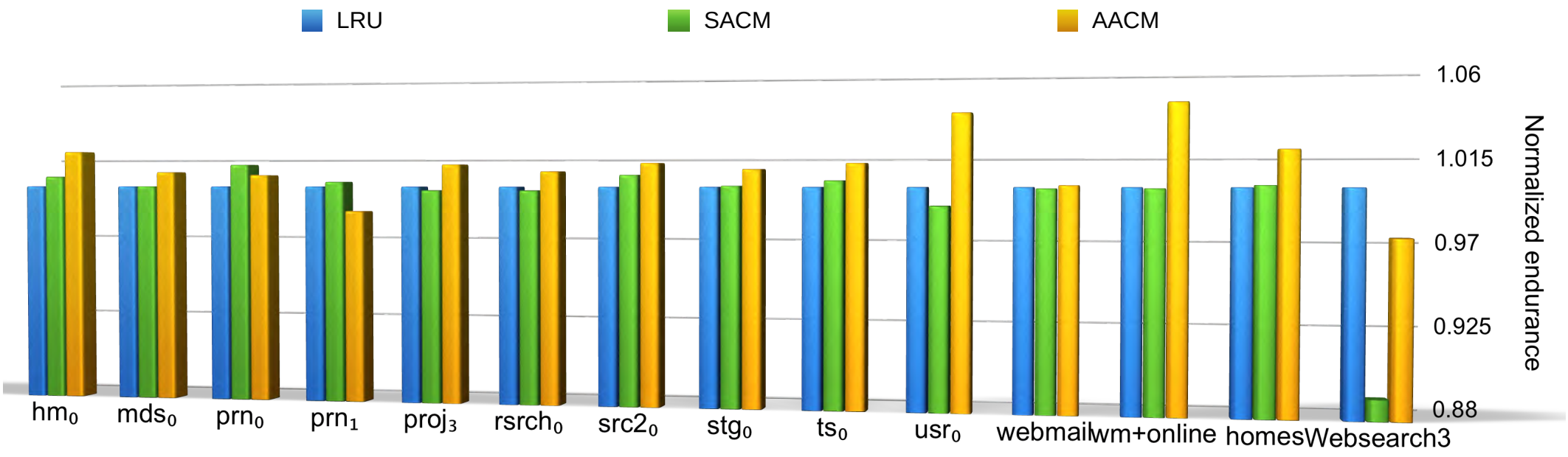
Evaluation : Endurance

- **Endurance**
  - REF harms the endurance from refresh operations



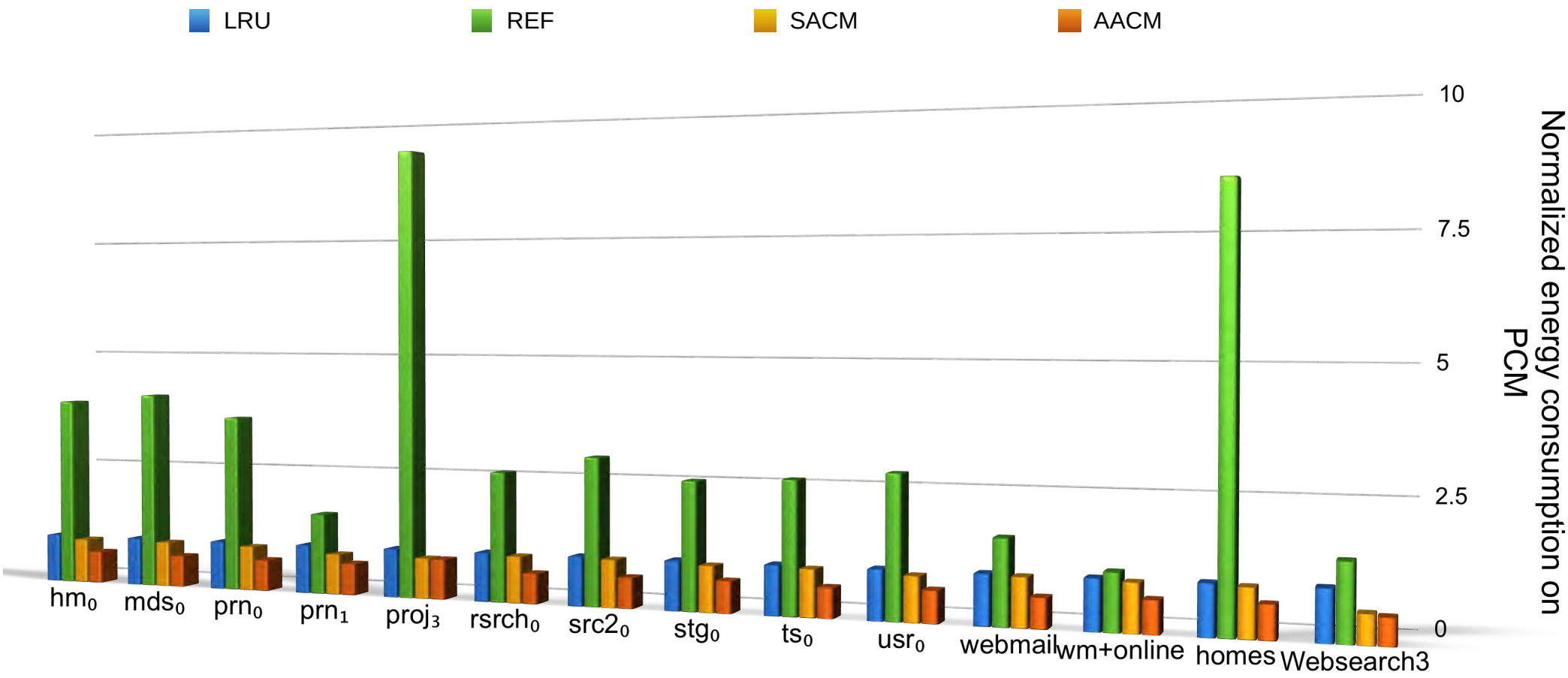
Evaluation : Endurance (without REF)

- **Endurance**
  - REF harms the endurance from refresh operations
  - SACM showing similar write counts to LRU
  - AACM incurs roughly 1% more writes compared to LRU (4% at maximum
  - Considering the MLC PCM endurance (10^5), the total amount of writes (wm+online), we can estimate that the lifetime is around 26 years.



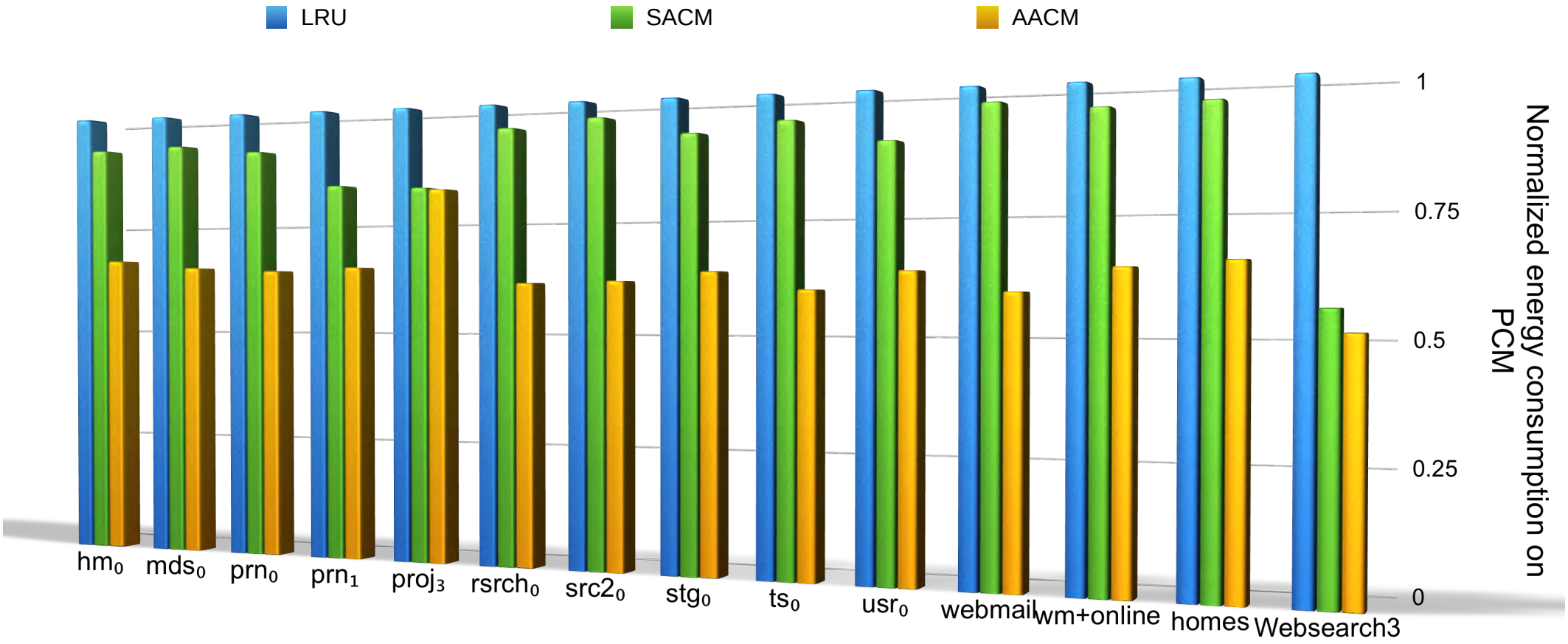
Evaluation : Energy consumption (PCM)

- **Energy consumption**
  - Energy = Nread x Energy-read + Nwrite x Energy-write
  - Adopt the energy model proposed by Liu et al., ASPLOS' 14
  - REF is 9 times higher than LRU (refresh overhead)



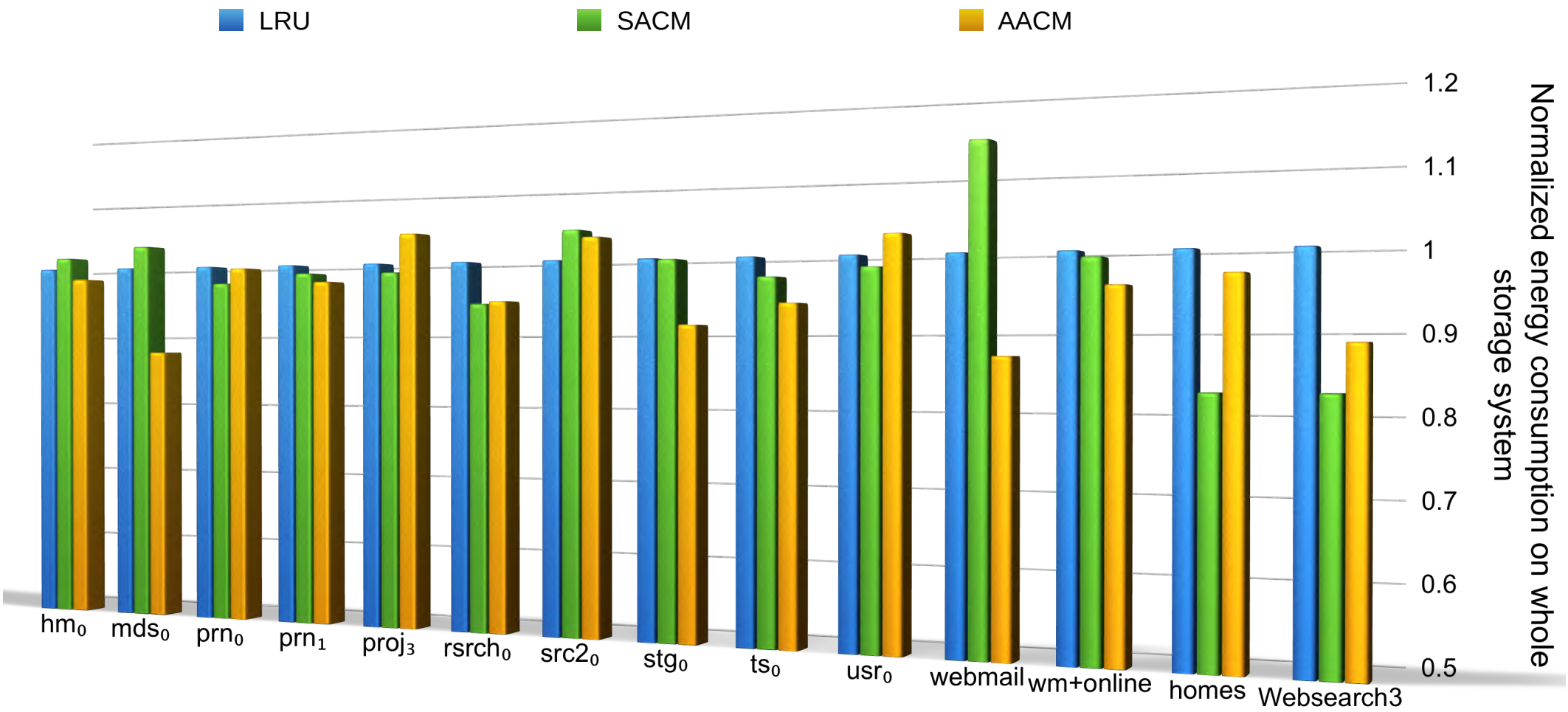
Evaluation : Energy consumption (PCM)

- **Energy consumption**
  - SACM reduces energy consumption on average 11%
  - AACM saves energy consumption on average 37% (and as high as 49%)



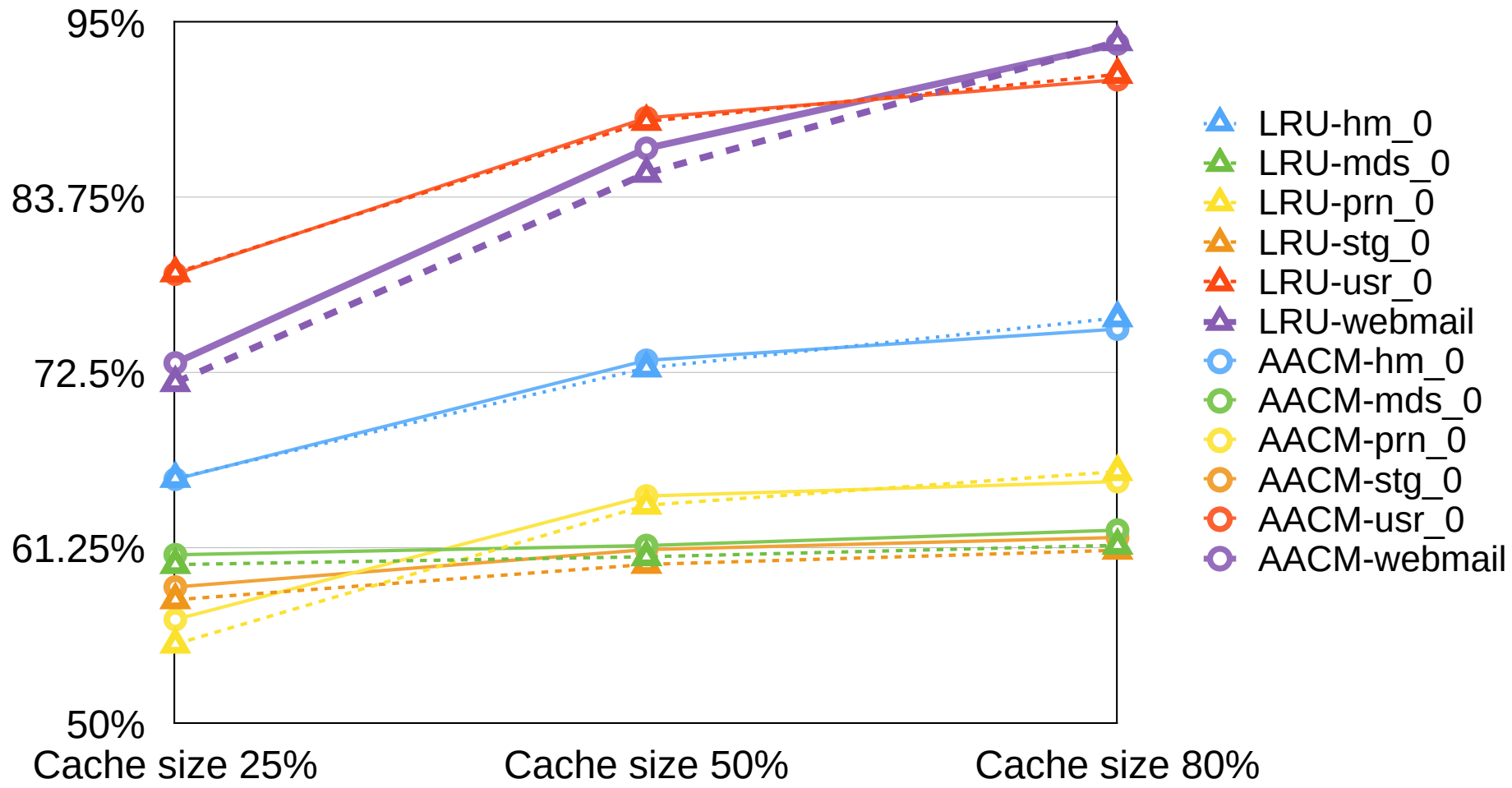
Evaluation : Energy consumption (whole storage system)

- **Energy consumption**
  - AACM saves energy by an average of 13% on whole storage system
  - Cause of retention relaxation and reduction of accesses in SSD



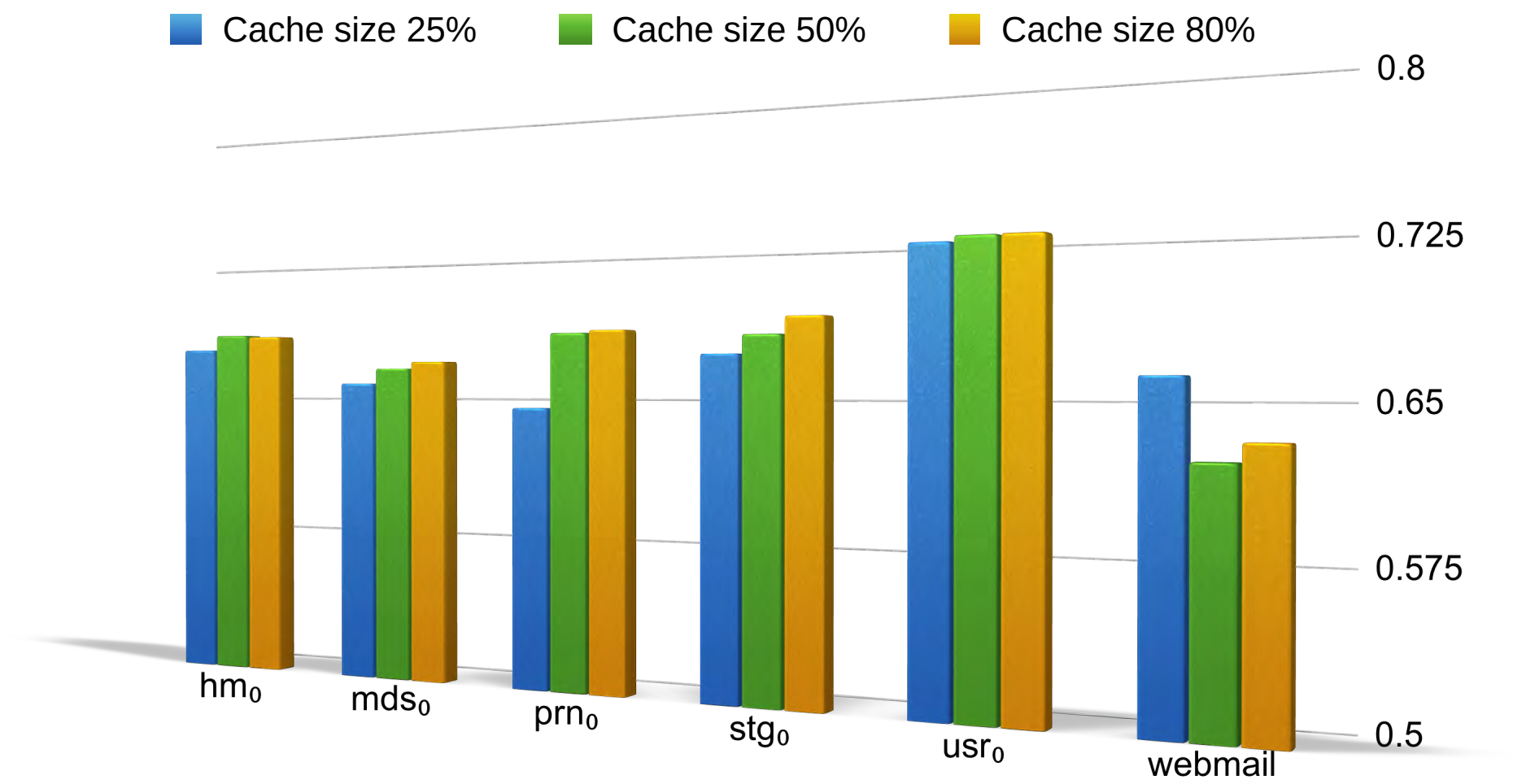
Evaluation : Hit ratio with various cache size

- **Hit ratio and latency with various cache size**
  - AACM performs better when the cache size is set to be small
  - Also, when the cache size becomes larger, both schemes show comparable performance since LRU also keeps most of the cacheable data



Evaluation : Latency with various cache size

- **Hit ratio and latency with various cache size**
  - In terms of latency, AACM outperforms LRU due to retention relaxation for all considered cache sizes



Outline

- Introduction & Motivation
- Design
- Evaluation
- Conclusion

Conclusion

- **Conclusion**
  - We suggest "**Amnesic notion**"
  - Exploit limited retention capability
  - Experimental results show that our proposal is effective in terms of performance and energy consumption.
    - AACM can reduce write latency by up to 40% (30% on average)
    - Also, AACM save energy consumption by up to 49% (37% on average)

Q&A