SMC: Smart Media Compression for Edge Storage Offloading

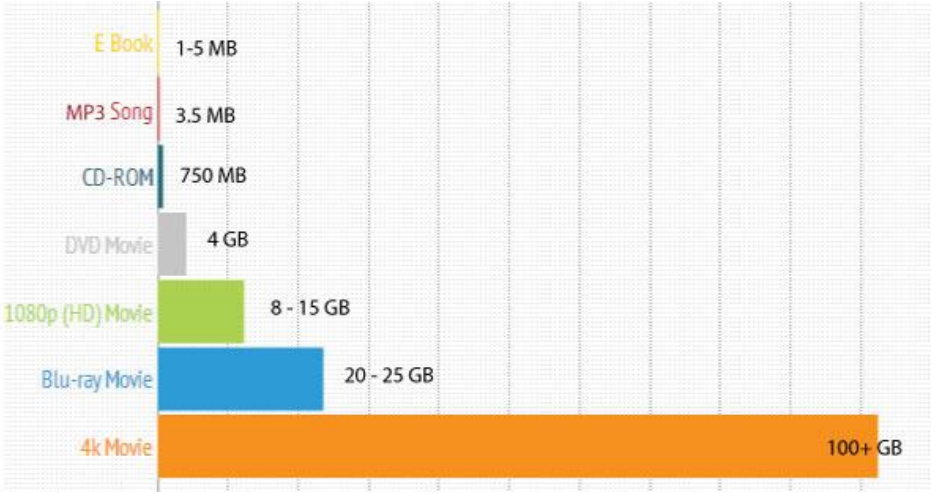
Ali Elgazar
Mohammad Aazam
Khaled A. Harras

General Outline

1) Problem description.
2) Proposed Solution.
3) Solution implementation.
4) Evaluation.
5) Conclusion.

Problem description (Files are too large)

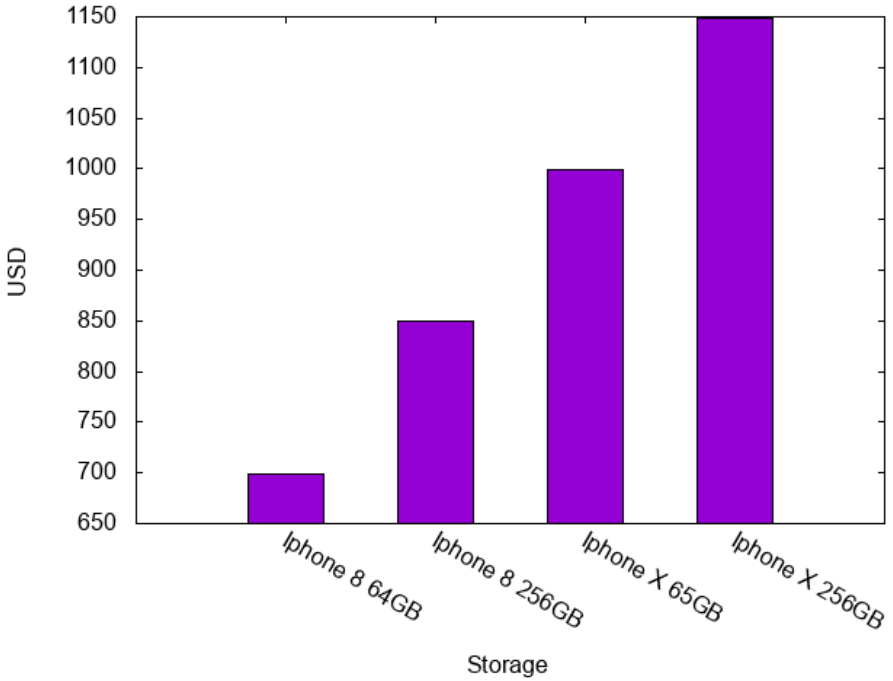
Rising media technologies and online presence results in large media files occupying large amounts of space on our device.



Problem description (Storage is too expensive)

Device storage is limited and higher storage capacity devices end up costing the user significantly more than what they would otherwise have to pay!

Or the user can rent storage on a cloud for 9.99$ a month!



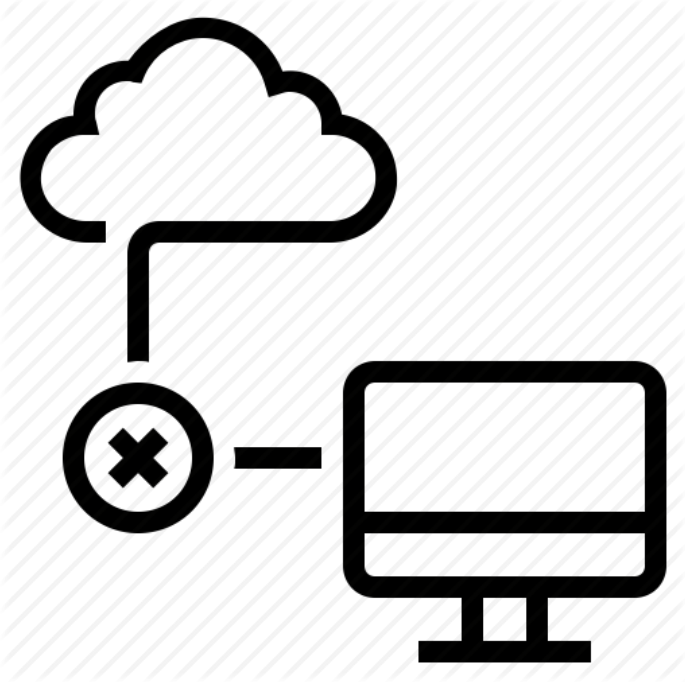
Problem description (Cloud is compromised)

Recent hackings of different cloud storage solutions lead to growing concerns over privacy and cloud security.

Problem description (Network infrastructure)

With low bandwidths/intermittent connections, offloading large quantities of files to centralized clouds becomes infeasible.

Distributed Edge Clouds (DECs), however, rely on their distributed nature to combat privacy/security concerns, and are typically equipped for areas with challenged networks. But...



Problem description (Energy use in compression)

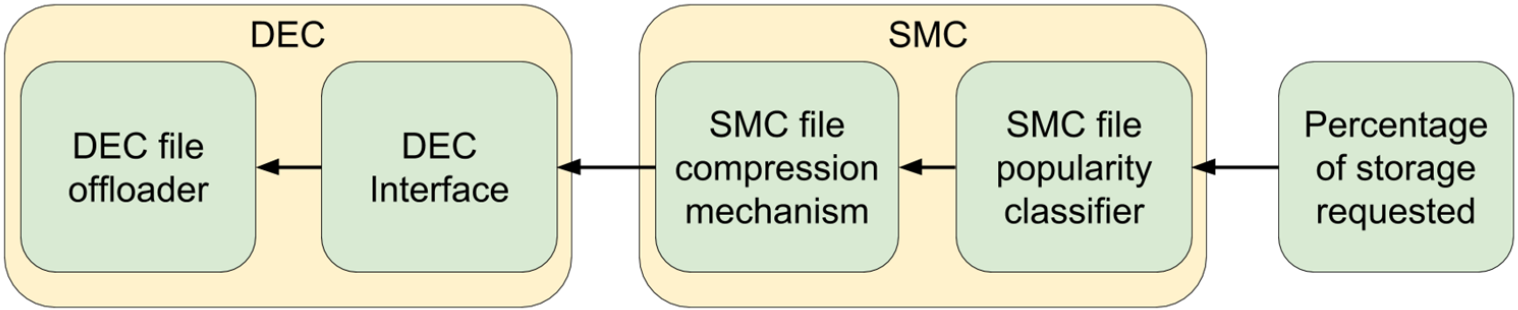
DECs lack optimizations based on compression. Compression is energy intensive on user mobile devices, as such, services tend to avoid compressing large volumes of media files, when offloading data.

General Outline

1) Problem description.
2) Proposed Solution.
3) Solution implementation.
4) Evaluation.
5) Conclusion.

Proposed solution (Smart Media Compression SMC)

SMC is an energy efficient compression and offloading mechanism, which is integratable with any DEC. SMC determines the relevance of files to the user by examining the user's file access patterns. SMC utilizes said relevance to automate offloading and compressing the user's files.



Proposed solution (SMC benefits)

By examining the user’s access patterns and classifying their files’ relevance, SMC provides three key benefits:

- Automating storage offload procedures.
- Reducing offloaded files through compression.
- Saving energy by selectively compressing only when necessary.

General Outline

1) Problem description.
2) Proposed Solution.
3) Solution implementation.
4) Evaluation.
5) Conclusion.

Solution Implementation (SMC operation)

SMC operates over several steps:

- Classifying file relevance.
- Determining necessity of compression.
- Determining compression ratios.

Solution Implementation (Classifying file relevance)

SMC classifies files into three categories based on their relevance to the user:

- Unpopular.
- Popular.
- Semi-Popular.

Solution Implementation (Classifying file relevance)

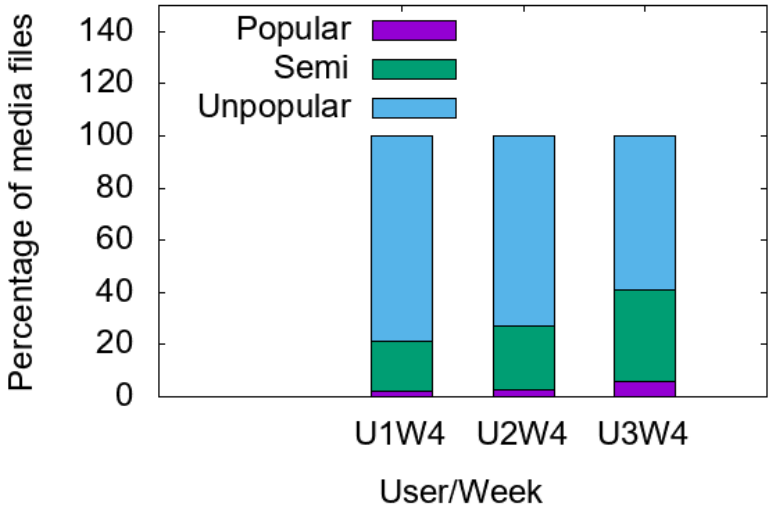
SMC augments a binary (popular/unpopular) algorithm called Pattern Based Popularity Assessment (PBPA). APBPA operates over several steps:

- Files not accessed recently (within a variable parameter) are considered stale and unpopular.
- The rest of the files are considered active files and the top accessed portions of said files are considered popular.
- Based on metrics established by PBPA, active files whose access patterns match the access patterns of popular files.
- The remaining active files are classified as semi-popular, based on temporal locality.

Solution Implementation (Determining necessity of compression)

Compression is only really required if there is a need to hold more files on the user's device.

This only occurs if the user requests an amount of storage offloaded that is greater than the amount of unpopular files.



Solution Implementation (Determining compression ratios)

The file compression ratio determines how much a media file will be compressed, and utilizes the following formulae.

$$CR_1 = \frac{TS - SR - PF}{SPF}$$ Ratio if compressing only semi-popular.

$$CR_2 = \alpha \times \frac{TS - SR}{PF + SPF}$$ Ratio if compressing both popular and semi-popular

Variable	Definition
TS	Total storage
SR	Storage requested
PF	Popular files
SPF	Semi-popular files

General Outline

1) Problem description.
2) Proposed Solution.
3) Solution implementation.
4) Evaluation.
5) Conclusion.

Evaluation (Setup)

- We attach SMC to a recently developed DEC, EdgeStore (ES)[1].
- We evaluate SMC+ES vs ES using a simulated environment running on the One simulator.
- We utilize a set of real life mobile device file access traces to simulate file accesses, and offloading[2].

[1] A. Elgazar, M. Aazam, and K. Harras, “Edgestore: Leveraging edge devices for mobile storage offloading,” (CloudCom) IEEE, 2018, pp. 56–61.
[2] R. Friedman and D. Sainz, “File system usage in android mobile phones,” in 9th SSC. ACM, 2016, p. 16

Evaluation (Parameter)

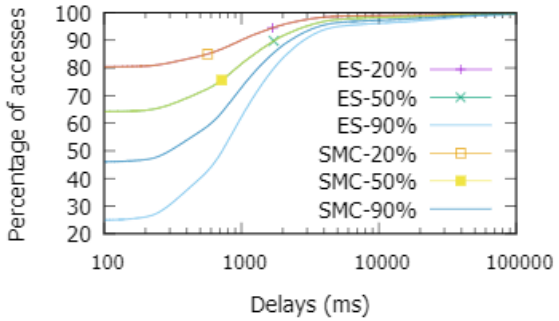
- Amount of storage offloaded: 20%, 50%, and 90%
- The platform used for offloading: EdgeStore Vs SMC+EdgeStore
- The environment in which the user resides: Metropolitan, Urban, Rural

Evaluation (Metrics)

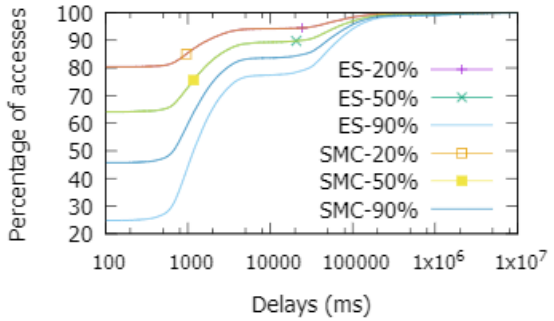
- Delay on accessing files: The amount of time from when a user accesses a file, to when it opens up on their UI.

- Battery consumption: The average hourly battery consumption of each platform during the simulation.

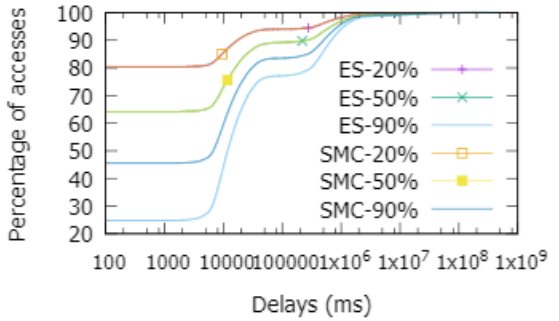
Evaluation (results)



(a) Metropolitan environment



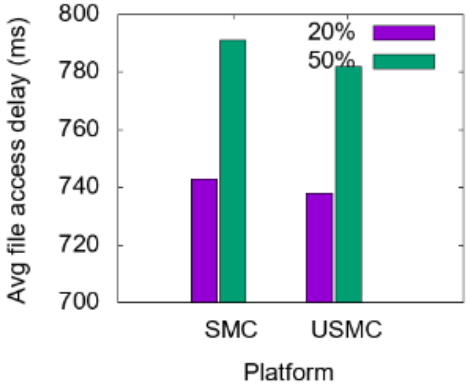
(b) Urban environment



(c) Rural environment

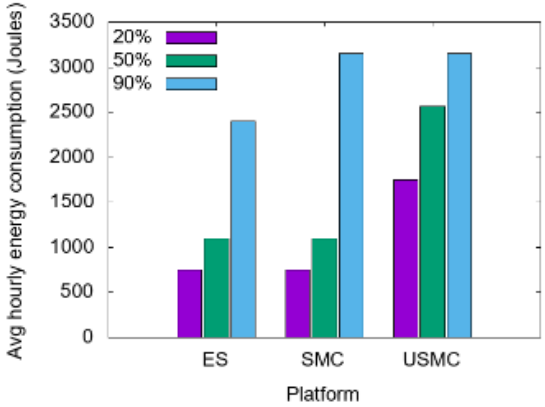
- On average, the user had 62%, 37%, and 4% unpopular, semi-popular, and popular files.
- Note that when we offload 50% or less of the user's files, ES and SMC perform the same, as no compression takes place due to the compression checks implemented.
- In the metropolitan, urban, and rural environments, the average delay drops from 312ms to 243ms, 432ms to 267ms, and 2.1s to 1.1s respectively, giving us a 28%, 61%, and 90% improvement in delays.

Evaluation (Removing the SMC checks)



- When SMC checks are removed, we can see that there’s almost no improvement in file access delays in the rural environment.

Evaluation (Impact on battery)



- The average hourly energy consumption in all simulations for ES, SMC, and USMC is1415, 1660, and 2490 Joules respectively.
- Devices running ES, SMC, and USMC ran out of battery every 33, 28, and 19 hours respectively on average.
- While SMC costs an additional 14%of energy consumption, it prevents an additional 43% energy consumption through its smart checks.

General Outline

1) Problem description.
2) Proposed Solution.
3) Solution implementation.
4) Evaluation.
5) Conclusion.

Conclusion (Overall message)

- Compressing the user’s media files can be utilized to not just lower network traffic, but also to dramatically enhance the average access time by retaining more files on the user’s device.
- We can determine and predict the relevance of files to the user in order to leverage such compression systems appropriately.
- Areas with challenged networks benefit the most from such systems.
- Large scale compression can be energy efficient when done selectively based on the user’s needs and access patterns.

Conclusion (Discussion topics)

- We'd like some feedback on our choice of attaching SMC to DECs instead of centralized clouds.
- A controversial point in this paper is that while we are providing a solution that reduces the potential energy consumption of compression, our solution still requires more energy consumption overall from any system it is attached to.
- In this paper, we don't examine or account for the networking conditions themselves when compressing files, this might lead to more optimal delays on accessing files depending on the environment.
- Our compression algorithm relies on accurate classification of the user's file relevance. If such classification was inaccurate for any reason, not only would the idea fall apart, but it would completely backfire.

Q&A time! Thank you!

We are here at the Carnegie Mellon Qatar Networking and Systems Lab have many interesting and ongoing projects! Check us out using the following code!