

# ハイブリッド型メインメモリ KVS サーバの提案及び エッジコンピューティングへの適用

小沢公基<sup>†</sup> 広瀬崇宏<sup>‡</sup> 高野了成<sup>‡</sup> 菅谷みどり<sup>†</sup>

芝浦工業大学工学部<sup>†</sup> 国立研究開発法人 産業技術総合研究所<sup>‡</sup>

## 1. 研究背景と目的

IoT (Internet of Things) 技術の進展により、スマートデバイスやセンサなどから大量のデータがサーバに転送され処理されている。今後ユーザ近傍のエッジ (エンドユーザとクラウドデータセンタとの中間層) においてもデータの処理が必要となる。このような考えはエッジコンピューティング技術[1] (Fog, cloudlet, MEC 等) として注目が集まっている。

エッジコンピューティングにおいては、より高い即応性を実現するためにエッジ側においてもデータを保持する必要がある。本研究では、大量のデータを一時的にキャッシュするために Key-Value-Store (KVS) サーバに注目した。KVS は大量のデータに対する低レイテンシでのアクセスを実現するために、大容量のメモリを搭載する必要があるが、現在メインメモリとして使われている DRAM の容量は制限がある。そこで不揮発性メインメモリ (NVMM) を用いて、キャッシュサーバを大容量化するとともにリクエストのレイテンシを改善する事が出来ると考え、この実現を研究の目的とした。

## 2. 課題

本稿では代表的なインメモリ KVS の Memcached[2] を対象とした KVS サーバを検討する。また NVMM として Intel Optane DC Persistent Memory (DCPM) [3] を対象とした。DRAM と NVMM の両方から構成されるハイブリッド型メインメモリでは、両者のメモリデバイスの使い分けが課題となる。NVMM を用いてメインメモリの容量を拡張し、キャッシュデータに対するヒット率を上げつつ、DRAM よりも低速な NVMM による性能の悪化を避けることが課題となる。

Memcached には DRAM に加えてストレージも利用してデータを保持する機能が既に存在している。しかし、書き込み待ちバッファでのバッファリングなど、DRAM よりも大幅に速度が劣るフラッシュストレージ等を想定した設計となっている

Proposal of hybrid main memory KVS server and its application to edge computing

Kouki Ozawa<sup>†</sup> Takahiro Hirofuchi<sup>‡</sup> Ryousei Takano<sup>‡</sup>

Midori Sugaya<sup>†</sup>

Faculty engineering, Shibaura Institute of Technology<sup>†</sup>

National Institute of Advanced Industrial Science and Technology<sup>‡</sup>

ことから、十分な性能を引き出すことはできないと予想される。そもそも DCPM は DRAM 同様にメインメモリとしてアクセスが可能であり、ストレージよりレイテンシが低い。これを考慮した DCPM に即した新たな仕組みが必要である。

## 3. 提案

メインメモリとして DRAM と同様にアクセスできる特性を考慮して、Memcached のメモリ管理機構をそのまま連続的に拡張し、(1) できるだけ既存の設計から差分を持たず、(2) アクセス頻度の違いにより状態を移動する設計を考えた。アイテムのアクセス頻度によって 2 種類のメモリ上のアイテムを動的に移動する仕組みを設けた。アクセスが多いアイテムを高速な DRAM に保持し、アクセスが少ないアイテムを低速な DCPM に保持する。これにより、大容量のデータ保持を可能としつつ、低速な DCPM を用いることによる性能低下を緩和する。

### 3.1. セグメント LRU アルゴリズムの拡張

Memcached は LRU を用いており、さらに大量のアイテムを効率的に管理するために、HOT, WARM, COLD の 3 つの状態を用いて、アイテムへのアクセス状態に応じて管理するセグメント LRU が用いられている。拡張に際して既存の DRAM 用セグメント LRU に加えて、新たに DCPM 用のセグメント LRU を設けた。2 つの LRU の移動の中間に MOVE キューとよばれる、DRAM と DCPM のメモリ

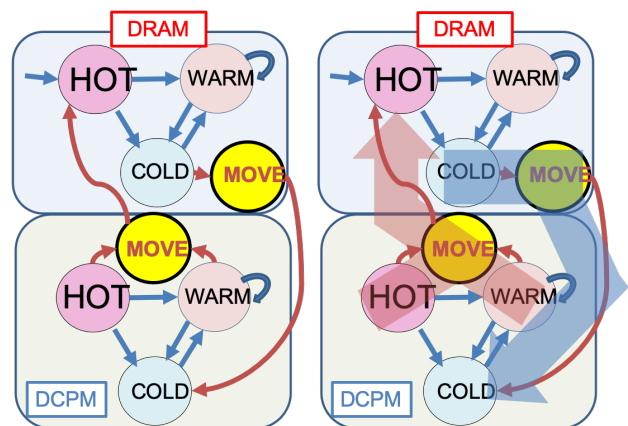


図1 拡張セグメント LRU

間移動の対象となったアイテムを追加するキューを作成し、両セグメント LRU を 2 つの MOVE キューによって連結する。図 1 左図に概要を示す。

図1右図では、実際に移動する経路を示した。アクセスの多いアイテムをDCPMからDRAMに移動する処理では、DCPMのHOTキューもしくはWARMキューに存在するアイテムをDRAMへの移動用のMOVEキューに追加した後にHOTキューにそのアイテムを移動する。アクセスが少ないアイテムがDRAMに長く存在しメモリを浪費しないように、DRAMのCOLDキューの末尾のアイテムが、設定した時間を超過している場合、DCPMにアイテムを退避する。一旦MOVEキューに移動した後に、DCPMへの移動させる。直接メモリにアクセスすることでメモリ間の移動を行うため、既存手法よりも高速に動作する。またセグメントLRUをDCPM用に拡張したことで、2種類のメモリデバイスに加えて既存のext\_storeを用いてストレージを使用し3層に拡張することも可能である。

#### 4. 評価実験

メモリデバイスによるKVSサーバの影響、提案手法と既存手法の比較、エッジにおける拡張KVSサーバの有用性を確認するためにレイテンシを計測する評価実験を行った。

##### 4.1. 実験環境・比較対象

表1 比較対象

DRAM-ONLY	DRAM 32 GB
DCPM-ONLY	DCPM 32 GB
提案-移動なし	DRAM 4 GB , DCPM 28 GB
提案-移動あり	DRAM 4 GB , DCPM 28 GB
EXTSTORE	DRAM 4 GB , DCPM 28 GB

表2 実験環境

	クライアント, クラウド	エッジ
CPU	Intel Xeon CPU E5-2630 2.20 GHz	Intel Xeon Gold 6230 2.10 GHz
メモリ	125 GB	186 GB
OS	Ubuntu18.04.03	Fedora30

##### 4.2. メモリデバイスによるKVSサーバへの影響、提案手法と既存手法の比較

MemcachedのクライアントプログラムであるMemslapを使用した。クライアントからエッジサーバへの同時接続数は256とした。使用した計算機は表2に示す。

##### 4.3. エッジにおける拡張KVSサーバ

クライアントはまずエッジサーバにアクセスしデータがなければクラウドサーバにアクセスする。クラウドの物理的な距離を想定してクライアントクラウド間の通信に1msの遅延を入れた。

##### 4.4. 評価結果

###### 4.4.1. メモリデバイス毎、従来手法との比較

評価結果からメモリデバイスのアクセス性能がKVSサーバのレイテンシに影響があることが分か

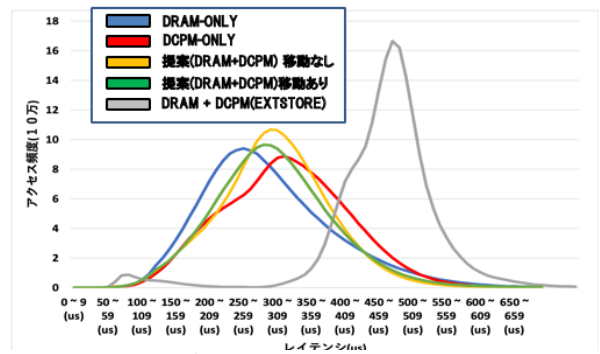


図2 メモリデバイス毎、従来手法との比較結果

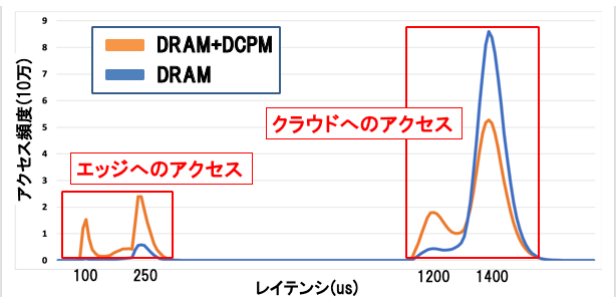


図3 エッジ、クラウドの評価

る。アクセス頻度の移動により、DRAMへのアクセスが増えることでレイテンシが改善した。従来手法のext\_storeに比べて提案手法はレイテンシが4割改善した。

###### 4.4.2. エッジにおける拡張KVSの評価結果

DCPMでエッジサーバを拡張することで、エッジへのアクセスが約5倍となった。

#### 5. まとめ

DCPMの実機を用いてKVSサーバを拡張し、アクセス頻度によるメモリデバイスの使い分けを実装し、評価を行った。メモリデバイスによるサーバの性能への影響と、提案により性能低下が軽減すること、拡張KVSサーバはエッジコンピューティングのデータキャッシュ部分に貢献できることが分かった。

##### 謝辞

本研究の一部はJST/CREST JPMJCR19K1の支援を受けた。本研究の一部は科研費19H01108の助成を受けた。

##### 参考文献

- [1] Kasif Bilal, Osman Khalid, Aiman Erbad, Samee U. Khan. Potentials, trends, and prospects in edge technologies: Fog, cloudlet, mobile edge, and micro data centers. Computer Networks. 2018, vol.130, p.94-120.
- [2] dormando. Memcached wiki. <https://github.com/memcached/memcached/wiki>, 2019.
- [3] インテル® Optane™ DC パーシステント・メモリー. <https://www.intel.co.jp/content/www/jp/ja/architecture-and-technology/optane-dc-persistent-memory.html>, (参照 2019-11-7).