

# 複数 VM 環境における KVS 性能の向上に関する一考察

徳田 大輝<sup>†</sup> 御代川 翔平<sup>†</sup> 山口 実靖<sup>†</sup>

工学院大学大学院工学研究科 電気・電子工学専攻<sup>†</sup>

## 1. はじめに

クラウドコンピューティングの普及により分散データベースが重要視されており、その一つとして KVS (Key-Value Store)がある。KVS は、データ構造の簡素化と一貫性保証の程度を下げることでスケーラビリティの向上を図っており、サーバ増設による性能向上と耐障害性向上を実現している[1]。KVS は仮想計算機を用いたクラウドコンピューティング環境で実行されることも多く、仮想化環境における性能が重要であると考えられる。

本研究では、KVS の一つである Cassandra と、仮想計算機システムの一つである KVM に着目し、複数 VM 環境における KVS の性能向上手法について考察する。

## 2. KVS (Key-Value Store)

KVS は、Key と Value の組を書き込み、Key を指定することで Value を得ることができるデータベース管理ソフトウェアである。代表的な KVS の実装に Cassandra [2]がある。

Cassandra はオープンソースの KVS であり、BigTable [3]のデータモデルと Dynamo [4]の分散ハッシュテーブルを併せ持った分散データベース管理システムである。Cassandra はハッシュ法を使用して各ノードにトークンと呼ばれる値を割り当て、Key と Value の組を各ノードに割り当てる。この時、トークン値の担当範囲をノード別に指定することで、各ノードに公平にデータ量を分散することができる。また、耐障害性の高さ、ノードの非集中性、高可用性、動的に伸縮可能なスケーラビリティ、設定可能な一貫性などの機能を持っている。

## 3. 仮想計算機

### 3.1. KVM (Kernel-based Virtual Machine)

一般にクラウド環境などは仮想計算機を用いて構築される。KVM は代表的な仮想化システムの一つであり、本研究では KVM を用いて調査を行う。KVM は Linux カーネル内に実装されており、OS をハイパバイザとして稼働する。

本研究では、仮想 HDD はイメージファイルモードを使用し、同モードではゲスト OS 上のアプリケーションはゲスト OS ファイルシステム、仮想計算機、ホスト OS ファイルシステムを介して HDD へのアクセスが行われる。

### 3.2. KSM (Kernel Same-page Merging)

KSM は各 VM が利用しているページの内、同一の内容のメモリページを1つの物理ページにまとめる機能で、実際に搭載しているメモリ量よりも見かけ上多くのメモリを仮想マシンで利用することができる。また、KSM がスキャンを積極的に高速で行うか、低速で行うか、停止するかなど状況に応じて調整するために ksmtuned というデーモンが用意されている。ksmtuned のスキャン停止時間は次の式で定義されている。

$$\text{sleep} = \frac{X \times 16 \times 1024 \times 1024}{\text{memory\_size}}$$

X は設定内の KSM\_SLEEP\_MSEC の値によって決まるパラメータであり、本稿では ksmtuned の初期の停止時間は 168[s]となっている。

## 4. 仮想化環境における Cassandra の性能評価

本章にて、仮想化環境における Cassandra 性能の評価を行う。

### 4.1. 測定環境

ホスト OS(物理計算機)数、ゲスト OS(仮想計算機)数と YCSB(Yahoo! Cloud Serving Benchmark)[5]で得られる性能の関係を評価した。Cassandra はレプリカ数を1とし、データベースサイズは 8[GB]、トークン範囲は均等に割り当てた(例: ノード数 6 の場合、トークン範囲は各ノードにつき 1/6)。2 台のホスト OS を使用し、各ホスト OS で 1 から 3 台までのゲスト OS を稼働させ、計 6 台まで使用した。また、YCSB をクライアント PC 上で実行し、ゲスト OS で稼働する Cassandra に対して負荷をかけた。YCSB のスレッド数は 18 スレッド、read/write 比率は 50%/50%とし、リクエスト分布方式は一様分布とした。使用したホスト OS とゲスト OS の仕様は表 1 の通りである。

表 1. 使用計算機の仕様

|        | Host OS                | Guest OS                    |
|--------|------------------------|-----------------------------|
| OS     | CentOS6.5              |                             |
| Kernel | Linux 2.6.32.27        | Linux 2.6.32-431.el6.x86_64 |
| Memory | 4[GB]                  | 1[GB]                       |
| CPU    | AMD Turion II Neo N54L |                             |

### 4.2. ホスト OS 2 台における性能評価

各ホスト OS 上のゲスト OS を 1 台、2 台、3 台と増やし、ゲスト OS 上における Cassandra 性能を評価した。測定結果を図 1 に示す。図 1 より、ゲスト OS 数が増えたとスループットが減少することがわかる。これは 1 つの HDD に複数の VM イメージファイルを配置すると、巨大なイメージファイル間での長距離シークにより、I/O 性能が低下することが原因と考えられる。1 秒当たりの長距離シーク回数を図 2 に示す。これは 50GB 以上 HDD シーク移動するものを長距離シークとみなし、1 秒当たりの回数を表したものである。図 2 より、VM の数が多いほど 1 秒当たりの長距離シークが発生していることがわかる。

### 4.3. KSM を用いたときの性能評価

次に各ホスト OS 上でゲスト OS を 3 台起動し、計 6 台の VM 上で Cassandra を起動させ、初期状態、KSM のみ実行した場合、KSM と ksmtuned を適用した場合の 3 パターンの性能の比較を行った。結果を図 3 に示す。図 3 より、KSM を適用した場合、初期設定よりもスループットが高くなることが確認された。これは、すべてのゲスト OS 上で同じアプリケーションが起動しているため、KSM によるページマージングが働き、物理ページの重複が減ることによってホスト OS のメモリキャッシュが増えたことが原因だと考えられる。また、ksmtuned を適用することで通常の KSM

A Study on the improvement of KVS performance in multiple VM environment

<sup>†</sup>Taiki Tokuda<sup>†</sup>, Shouhei Miyokawa<sup>†</sup>, Saneyasu Yamaguchi<sup>†</sup>

<sup>†</sup>Electrical Engineering and Electronics, Kogakuin University Graduate School

よりも高いスループットが確認された。スキャン間隔の延長が確認されており、スキャン間隔の調整により性能が向上したのだと考えられる。また、VM数毎のKSM適用後の性能比較を図4に示す。図4より、VM数が少ないほどKSMによる性能向上が大きいことがわかる。

以上から、スキャンの間隔を変更することで性能向上が可能であることが分かる。通常のksmtunedの実行内容であるKSMスキャン間隔を1/2倍,1倍(通常),2倍と変更し、6台のVM環境にて実験を行った。結果を図5に示す。図5より、スキャン間隔を詳細に調整するとさらに高いスループットを得ることが分かる。これはスキャン間隔が短いとKSMで多くのページがスキャンされ多くのページがマージされ、結果としてより多くのメモリがVMに与えられたためであると考えられる。

5. まとめ

本稿では、仮想化環境におけるKVSの性能に着目し、性能評価を行った。評価より、仮想計算機の数が多いほど性能が低下することが確認された。また、複数VM上で同一のアプリケーションが実行されている点に着目し、KSMの適用について考察を行った。評価の結果、KSMを用いて同一ページのマージを行うことにより性能が向上することが確認された。

今後は、KSMの改良によるCassandra性能のさらなる向上についての考察を行う予定である。

謝辞

本研究はJSPS科研費24300034,25280022,26730040の助成を受けたものである。

参考文献

- [1] 堀内 浩基,山口 実靖,“KVSにおける動的性能拡張性の向上”, 研究報告マルチメディア通信と分散処理(DPS-154, 03, 2013)
- [2] Avinash Lakshman and Prashant Malik, “Cassandra- A Decentralized Structured Storage System”, LADIS 09, 2009
- [3] Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kakulapati, Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Vosshall and Werner Vogels, “Dynamo: Amazon’s Highly Available Key-value Store”, SOSP ’07, 2007
- [4] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes and Robert E. Gruber, “Bigtable: A Distributed Storage System for Structured Data”, IOSDI ’06 pages 205--218, 2006
- [5] Brian F. Cooper, Adam Silberstein, Erwin Tam, Raghu Ramakrishnan, Russell Sears “Benchmarking Cloud Serving Systems with YCSB”

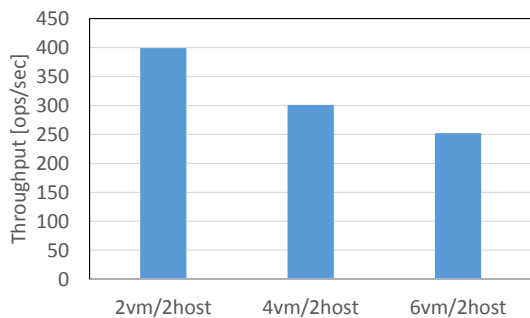


図1 VM数とCassandra性能の関係

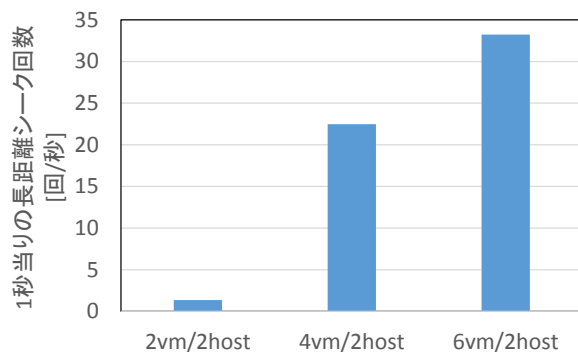


図2 VM数と長距離シーク回数の関係

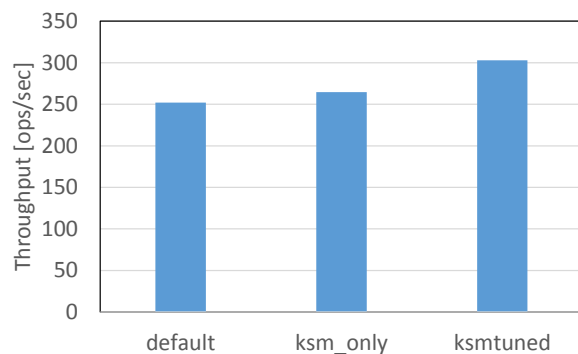


図3 KSM, ksmtuned適用時のCassandra性能の評価

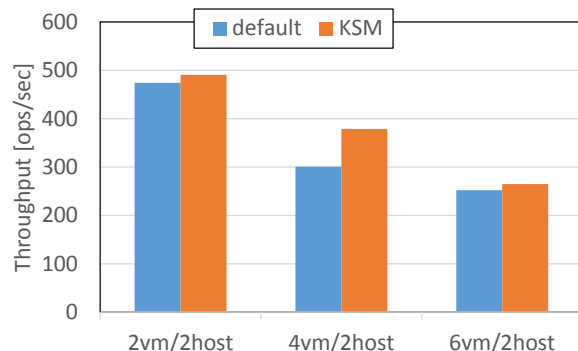


図4 VM数毎におけるKSM適用時のCassandra性能の評価

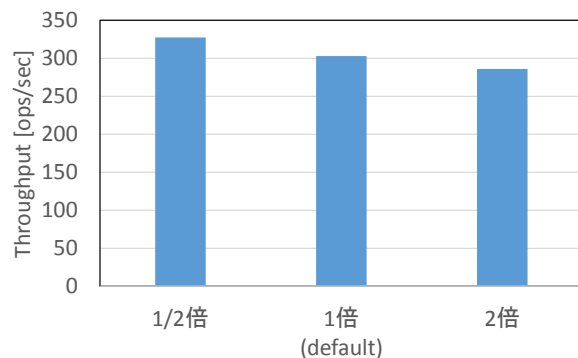


図5 スキャン間隔を変更した時のCassandra性能の評価