STUDIA INFORMATICA 2012
Volume 33 Number 3A (107)

Dariusz CZERWIŃSKI
Politechnika Lubelska, Wydział Elektrotechniki i Informatyki

WPŁYW ZARZĄDCY MASZYNY WIRTUALNEJ NA WYDAJNOŚĆ SYSTEMÓW EKSPLORACJI DANYCH

Streszczenie. Artykuł prezentuje analizę porównawczą wpływu zarządcy maszyny wirtualnej na wydajność systemów eksploracji danych. Dyskusja opiera się na wynikach otrzymanych w środowisku testowym opartym na dystrybucji Cloudera Hadoop pod kątem wykorzystania go jako prywatnego klastra na komputerach klasy desktop. Główny nacisk został położony na wpływ wybranych, popularnych hypervisorów na typowe operacje, takie jak obliczenia zrównoleglone, odwołania do pamięci oraz wykorzystanie zasobów CPU.

Słowa kluczowe: wydajność systemu, wirtualizacja, systemy eksploracji danych, Cloudera Hadoop

INFLUENCE OF THE VIRTUAL MACHINE MANAGER ON THE DATA MINING SYSTEM PERFORMANCE

Summary. This paper presents a comparative analysis of the impact of the virtual machine manager on the data mining systems performance. Discussion is based on the results obtained in a test environment based on the Cloudera Hadoop distribution which is used as personal cluster. The main focus is the hypervisor impact on the typical operations in data mining system, such as parallelized calculation, memory operations and the use of CPU resources.

Keywords: systems efficiency, virtualization, data mining systems, Cloudera Hadoop

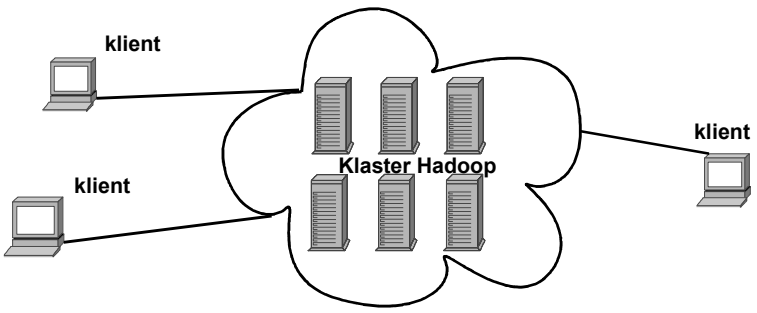
1. Wprowadzenie

Systemy eksploracji danych (ang. *data mining*) pełnią bardzo istotną rolę w branży IT. Są używane w wielu dziedzinach związanych z nauką, jak również znajdują zastosowania

w rozwiązaniach komercyjnych. Przykładem takiego systemu eksploracji danych jest projekt Hadoop rozwijany przez fundację Apache [4]. Projekt Hadoop jest częścią składową dystrybucji Cloudera i w jego skład wchodzi oprogramowanie oraz zestaw bibliotek umożliwiających tworzenie zadań z użyciem paradygmatu Map Reduce [2]. MapReduce jest w istocie użytecznym narzędziem do wykonywania analizy danych w chmurach obliczeniowych. Jednym z podstawowych założeń, jakie zostało przyjęte przez twórców projektu Hadoop, jest prostota programowania metod eksploracji danych. Pozwala to na wygodną implementację szerokiej gamy algorytmów przydatnych w wielu zastosowaniach. Aplikacje napisane w języku Java mają bezpośredni dostęp do Hadoop API, ale Hadoop posiada również funkcje, które pozwalają na implementację metod eksploracji w dowolnym języku programowania [1]. W chwili obecnej wiele organizacji, które potrzebują przetwarzać bardzo duże ilości danych, wykorzystuje w tym celu MapReduce. Firma Google jako pierwsza zaczęła używać MapReduce przed 2004 r. [2]. Firma Yahoo posiada największy klaster Hadoop, który wykorzystuje ponad 11 000 rdzeni [3]. Amazon wykorzystuje Hadoop jako część ich usług przetwarzania w chmurze. Hadoop jest używany między innymi przez takie podmioty, jak: Facebook, last.fm, New York Times, AOL Advertising, NAVTEQ, Samsung, TrendMicro i wiele innych, których listę można znaleźć na witrynie projektu Hadoop [4].

Zdecentralizowane obliczenia są szeroko stosowane w branży IT, a ich realizacje są niezwykle zróżnicowane. Na ich tle cechami wyróżniającymi projekt Hadoop są: [5]

- dostępność – Hadoop może działać tak na dużych klastrach ogólnie dostępnych maszyn, w chmurze obliczeniowej, takiej jak Amazon Elastic Compute Cloud (EC2) i małych prywatnych klastrach czy chmurach,
- solidność – Hadoop jest przeznaczony do pracy na komputerach klasy PC i wobec tego został zaprojektowany przy założeniu częstych awarii sprzętu,
- skalowalność – Hadoop charakteryzuje się liniową skalowalnością w sytuacjach, gdy zachodzi potrzeba zwiększenia ilości węzłów w klastrze,
- prostota – Hadoop pozwala użytkownikowi w prosty i efektywny sposób implementować metody zrównoleglające obliczenia.



Rys. 1. Idea działania klastra Hadoop, usługi dostępne dla klientów analogicznie jak w chmurze
Fig. 1. Idea of the Hadoop cluster, services available to clients analogous to the cloud availability

Ideę działania klastra Hadoop, który realizuje swoje zadania w chmurze obliczeniowej, przedstawia rys. 1. W najprostszym przypadku, klaster Hadoop to zestaw komputerów tworzących dedykowaną sieć komputerową, zazwyczaj zrealizowaną w jednej lokalizacji. Jest też możliwe skonfigurowanie klastra Hadoop tak, aby maszyny znajdowały się w różnych lokalizacjach. Przechowywanie i przetwarzanie wszystkich danych jest realizowane w „chmurze" maszyn. Różni użytkownicy zarówno w lokalizacjach zdalnych, jak i lokalnych mogą zgłaszać zadania do zarządcy klastra Hadoop.

2. System Cloudera

System Cloudera pozwala na budowę struktur zarówno klastrów obliczeniowych, jak również chmur prywatnych. Dzięki temu użytkownicy tego systemu korzystają z zasobów w ten sam sposób jak z zasobów chmury publicznej oferowanej przez Amazon. Pozwala to na prostą integrację tego rozwiązania w ramach projektów chmur hybrydowych [6]. Hadoop, który wchodzi w skład dystrybucji Cloudera, składa się z następujących elementów [4, 5].

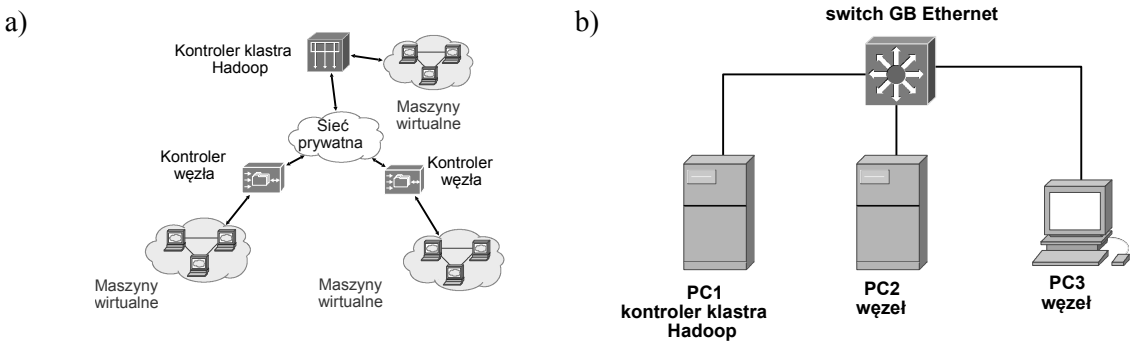
- Węzła podrzędnego SN (ang. *Slave Node*) – jest to zasób fizyczny (najczęściej pojedynczy host), na którym są uruchamiane zadania do wykonania. Węzeł ten może pracować zarówno w trybie gromadzenia danych (ang. *DataNode*), jak i trybie akceptowania zadań (ang. *TaskTracker*). Możliwe jest skonfigurowanie węzłów tak, aby pracowały one wyłącznie jako węzły gromadzące dane lub jako węzły obsługujące zlecona zadania.
- Węzła nadrzędnego MN (ang. *Master Node*) – w niewielkich klastrach urządzenie to pełni cztery funkcje: węzła, który gospodaruje zadaniami (ang. *JobTracker*) w klastrze, węzła akceptującego zadania (ang. *TaskTracker*), węzła utrzymującego drzewa katalogów w systemie plików HDFS (ang. *NameNode*), ale nie przechowuje danych oraz węzła przechowującego i replikującego dane w systemie HDFS (ang. *DataNode*).
- Systemu plików HDFS (ang. *Hadoop Distributed File System*) – jest elementem niezbędnym do przechowywania danych w klastrze. HDFS jest bardzo odporny na uszkodzenia, dzięki czemu może być wdrażany nawet na sprzęcie klasy PC. System plików HDFS zapewnia dostęp o wysokiej przepustowości do danych aplikacji.

2.1. Platforma testowa

Opracowana platforma testowa ma na celu ocenę wydajności systemu eksploracji danych w strukturze klastra prywatnego. Klaster ten składał się z trzech komputerów klasy PC, na których zainstalowano 64-bitowy system operacyjny klasy desktop. Badania przeprowadzone na podstawie tak zbudowanego klastra prywatnego są realizowane w dwu etapach. Pierwszy

etap miał na celu ocenę wpływu hypervisora na wydajność systemu eksploracji danych w sytuacji, gdy systemem operacyjnym gospodarza był Windows 7 Ultimate x64. Wyniki tego etapu są zaprezentowane w dalszej części artykułu. W kolejnym etapie autor planuje przeprowadzenie testów dla innych systemów gospodarza pochodzących z rodziny desktop Linux (OpenSUSE 12.x x86_64, Ubuntu 11.x x86_64). Taki plan badań ograniczył zakres wyboru realizacji wirtualizacji tylko do tych rozwiązań, które są wspólne zarówno dla systemów Linux, jak i MS Windows, wspierają 64-bitowe systemy gości i znajdują zastosowanie w rozwiązaniach komputerów klasy desktop.

Biorąc pod uwagę powyższe założenia, jako podstawę realizacji wirtualizacji, a zarazem badań wpływu hipervisora maszyny wirtualnej na wydajność systemu eksploracji danych, wybrano trzy różne pakiety: pakiet VMware (realizowany na podstawie VMware Player v.4), pakiet VirtualBox (realizowany na podstawie VirtualBox 4.1.10 for Windows hosts amd64) oraz pakietu WinKVM z Qemu 1.0.1 (implementacja KVM dla systemów MS Windows). Elementy klastra zostały zainstalowane i skonfigurowane na komputerach klasy PC wyposażonych w dwurdzeniowe procesory firmy Intel oraz AMD ze wsparciem dla sprzętowej wirtualizacji, 4 GB pamięci RAM, pamięć dyskową o pojemności 250GB oraz interfejsy sieciowe Gigabit Ethernet. W tak przygotowanym środowisku, na każdym z komputerów, została uruchomiana maszyna wirtualna z system Centos 5.3. System ten, z kolei, zawierał implementację dystrybucji Cloudera's Distribution Including Apache Hadoop (CDH) w wersji CDH3U3 [7].



Rys. 2. Architektura środowiska testowego: a) logiczna, b) fizyczna
Fig. 2. Testbed architecture: a) logical, b) physical

Architektura środowiska testowego została przedstawiona na rys. 2. Poszczególne elementy struktury testowego systemu Hadoop zostały przypisane do pokazanego na rys. 2b sprzętu komputerowego w następujący sposób:

- kontroler klastra, węzeł nadrzędny MN – komputer PC1 (AMD Athlon X2 240, 2 rdzenie), 4 GB RAM, 250 GB HDD SATA, system gospodarza Windows 7 Ultimate x64, system gościa Cloudera (Centos 5.3 x64);

- węzeł podrzędny SN – komputer PC2 (AMD Phenom X2 555, 2 rdzenie), 4 GB RAM, 250 GB HDD SATA, system gospodarza Windows 7 Ultimate x64, system gościa Cloudera (Centos 5.3 x64);
- węzeł podrzędny SN – komputer PC3 (Intel C2Duo P8700, 2 rdzenie), 4 GB RAM, 250 GB HDD SATA, system gospodarza Windows 7 Ultimate x64, system gościa Cloudera (Centos 5.3 x64).

3. Wyniki pomiarów

Na podstawie przedstawionego wcześniej klastra prywatnego wykonano testy, mające na celu zbadanie wpływu zarządcy maszyny wirtualnej na wydajności systemu eksploracji danych. Pomiary zostały zrealizowane w trzech różnych konfiguracjach testowych oznaczonych jako „VMware", „Virtual Box" i „WinKVM", gdzie system Cloudera był uruchamiany odpowiednio w następujących maszynach wirtualnych: VMware, Virtual Box oraz WinKVM.

W czasie testów systemy operacyjne gospodarza jak i gościa nie były obciążone dodatkowymi procesami. Równocześnie, w wydzielonej sieci lokalnej nie występował żaden dodatkowy ruch sieciowy, poza tym który zachodził między węzłami klastra. W celu zbadania wpływu zarządcy dobranych zostało kilka rodzajów testów tak, aby uzyskać miarodajne wyniki wydajności poszczególnych rozwiązań. Wybrane testy to:

- klasyfikacja przykładowych danych na bazie algorytmu regresji logistycznej przy założeniu jak największej dokładności modelu (parametr AUC=1, ang. *area under the curve*) w środowisku Mahout. Realizacje tego testu ilustruje poniższe polecenie:

```
mahout trainlogistic --input danewe.csv --output model --target kolor
--categories 2 --predictors x y a b c --types numeric
--features 20 --passes 400 --rate 50
```

  Zadaniem klasyfikatora było określenie, w jakim kolorze jest punkt o dowolnych współrzędnych. Dane uczące (użyte w procesie trenowania klasyfikatora) pochodziły z pliku danewe.csv zawierającego następujące kolumny: x, y – współrzędne punktu w układzie kartezjańskim (wartości w przedziale 0-1), a – odległość od punktu (0, 0), b – odległość od punktu (1, 0), c – odległość od punktu (0.5, 0.5), kolor – kolor punktu (wartości 1 lub 2). Predykatorami były wartości zmiennych x, y, a, b, c, natomiast zmienną docelową stanowił kolor;
- benchmark Dhrystones (No Opt – brak optymalizacji w czasie kompilacji, Opt3 – włączona optymalizacja w czasie kompilacji);
- benchmark Linpack (No Opt – brak optymalizacji w czasie kompilacji, Opt3 – włączona optymalizacja w czasie kompilacji);
- benchmark pamięci RAMSMP (zapis i odczyt bloków integer o różnym rozmiarze).

Każdy z testów składał się z 50 prób pomiarowych, dla których zostały określone wartości średnie, odchylenia standardowe oraz przedziały ufności. W przypadku otrzymanych wyników założono, że populacja generalna ma rozkład normalny (N(m,σ)). Przedział ufności dla wartości średniej dany jest wówczas zależnością:

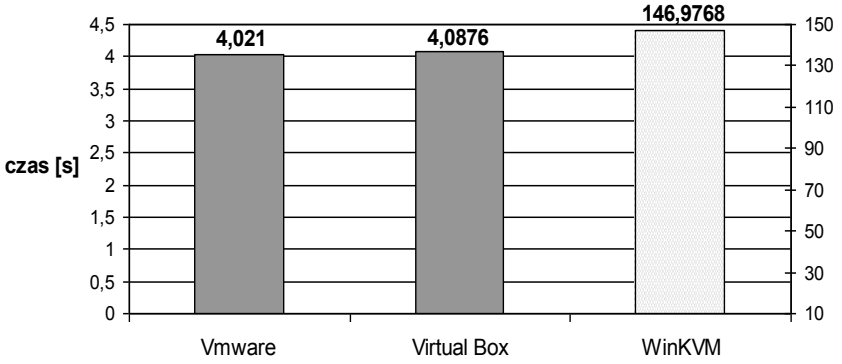
$$\bar{x} \pm \frac{u_\alpha S}{\sqrt{n}} \tag{1}$$

gdzie $\bar{x}$ – średnia arytmetyczna obliczona na podstawie n – elementowej populacji próby, S – odchylenie standardowe, u_α – wartość zmiennej losowej U o standaryzowanym rozkładzie normalnym (N(0,1)) wyznaczona w taki sposób, aby spełniona była relacja:

$$P\left\{\bar{x} - \frac{u_\alpha S}{\sqrt{n}} < m < \bar{x} + \frac{u_\alpha S}{\sqrt{n}}\right\} = 1 - \alpha \tag{2}$$

Przy obliczaniu przedziałów ufności został założony poziom istotności α = 0,02. Przedziały ufności zostały wyznaczone w celu udzielenia jednoznacznej odpowiedzi, czy wyniki pomiarów stanowią wyraźny trend wszędzie tam, gdzie przedziały *wartość średnia ± odchylenie standardowe* zachodziły na siebie (tabela 1 oraz tabela 2, wyniki platform VMware i Virtual Box dla benchmarku Dhrystones i RAMSMP).

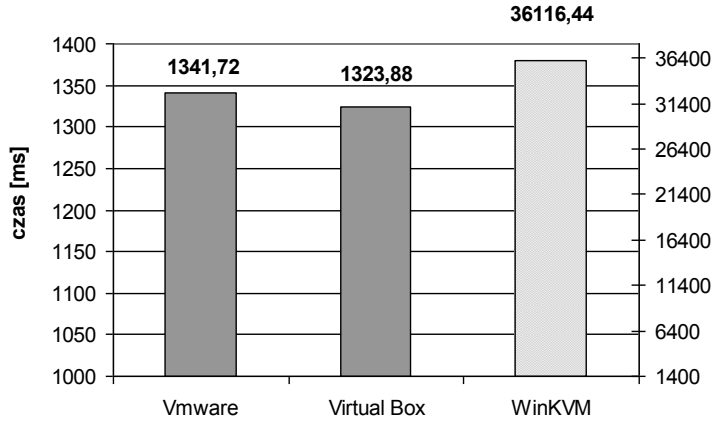
Wyniki testów uzyskane na podstawie wytrenowanego klasyfikatora (polecenie *mahout)* przedstawiają rys. 3 oraz rys. 4. Czas wytrenowania określony z użyciem polecenia *time mahout* przedstawia rys. 3, przy czym na wykresie zaprezentowano rzeczywisty przedział czasu, w jakim program był uruchomiony (ang. *real time*).



Rys. 3. Czasy wytrenowania klasyfikatora otrzymane z użyciem polecenia *time mahout*
Fig. 3. Classifier training times obtained with *time mahout* command

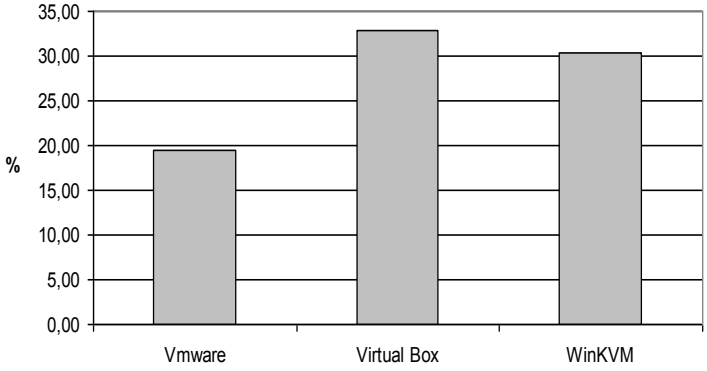
Wyraźnie widać, iż w maszynach wirtualnych VMware oraz Virtual Box wartości zmierzonego czasu są bardzo zbliżone, natomiast pomiary dla implementacji WinKVM znacznie odbiegają na niekorzyść od pozostałych dwu rozwiązań. Odchylenie standardowe dla VMware, Virtual Box oraz WinKVM było niewielkie i wynosiło odpowiednio: 0,1935, 0,1455 oraz 4,4123 sekund. Czas wytrenowania klasyfikatora, uzyskany z użyciem samego polecenia *mahout*, dla analizowanych rozwiązań hipervisorów, zestawiono na rys. 4. W tym przypadku

otrzymane wyniki również potwierdzają zależności otrzymane w poprzednim teście i wyniki dla hypervisora WinKVM są znacznie gorsze od pozostałych rozwiązań.



Rys. 4. Czasy wytrenowania klasyfikatora otrzymane z użyciem polecenia *mahout*
Fig. 4. Classifier training times obtained with *mahout* command

Zdecydowanie gorsze rezultaty otrzymane dla WinKVM są spowodowane tym, że projekt ten jest nadal na etapie rozwoju a implementacja wirtualizacji w dodatkowym module jądra systemu MS Windows nie jest jeszcze pełna.

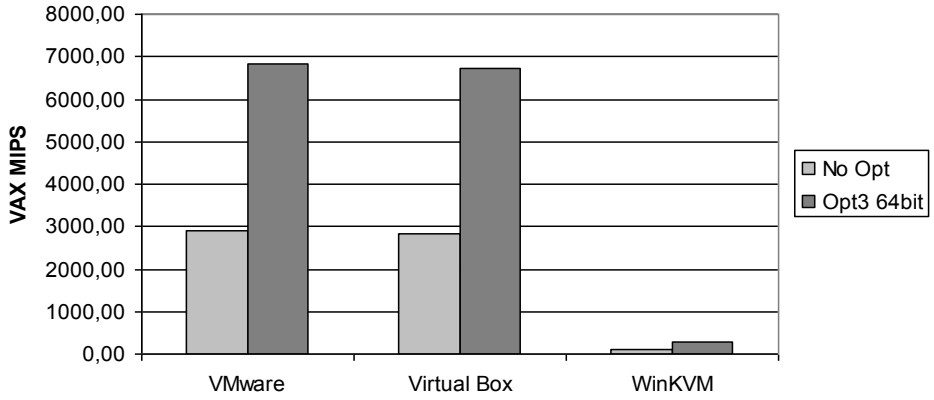


Rys. 5. Narzut systemowy dla testowanych zarządców
Fig. 5. System overhead for tested hypervisors

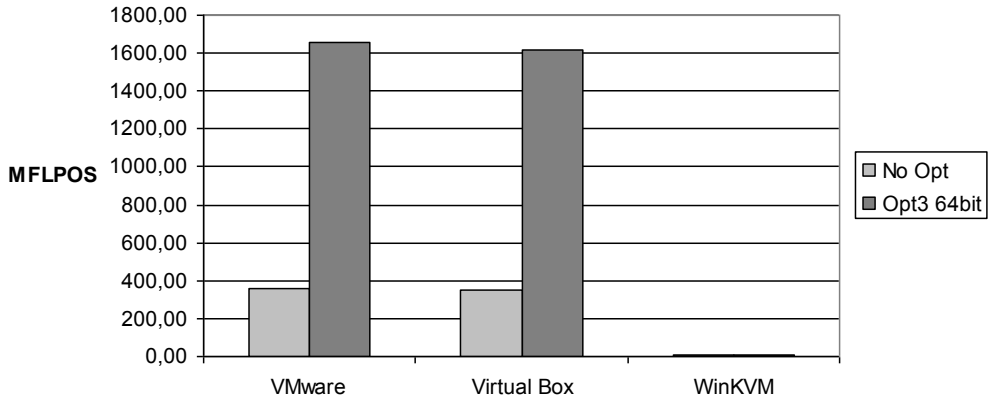
W celu uzupełnienia i pogłębienia przytoczonych wyżej wniosków ocenie został poddany dodatkowy parametr, tak zwany narzut systemowy. Narzut systemowy został zdefiniowany jako procentowa wartość, w postaci stosunku czasu pracy procesora w trybie systemu do czasu, jaki procesor rzeczywiście poświęcił na przetwarzanie procesu w trybie użytkownika (sys/user·100%). Wyniki porównania wartości narzutów systemowych dla poszczególnych maszyn wirtualnych przedstawiono na rys. 5. Można zauważyć, że najmniejszym narzutem cechowal się hypervisor Vmware, natomiast w przypadku hypervisorów Virtual Box oraz WinKVM narzuty te okazały się o ponad 10% wyższe.

Kolejnym benchmarkiem, wykorzystanym w trakcie testów prywatnego klastra, był Dhrystones. Na rys. 6 przedstawiono zestawienie otrzymanych wyników dla tego testu. Odchylenia standardowe i przedziały ufności zostały zestawione w tabeli 1. Na podstawie otrzymanych

wyników badań i ich analizy (przedziały ufności nie zachodzą na siebie) można stwierdzić, że najwyższa wydajność dla testu została uzyskana dla hypervisora VMware.



Rys. 6. Wyniki testu Dhrystones 2.1
Fig. 6. Results of Dhrystones 2.1 test



Rys. 7. Wyniki testu Linpack
Fig. 7. Results of Linpack test

Tabela 1

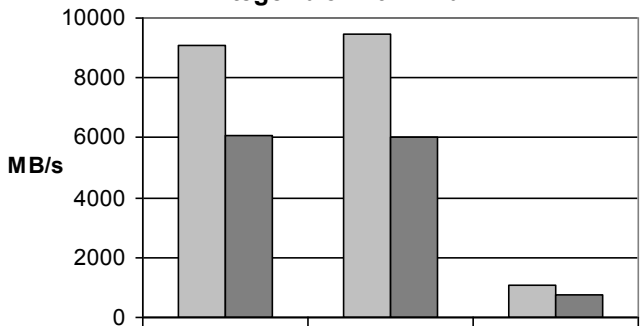
Zestawienie średnich wartości pomiarów, odchyleń standardowych oraz przedziałów ufności dla testu

Benchmark	Platforma	Optym.	Średnia	Odchylenie standard.	Przedział ufności
Dhrystones	VMware	No Opt	2894,60	51,58	2894,60±16,97
		Opt3 64 bit	6840,23	37,46	6840,23±12,32
	Virtual Box	No Opt	2840,16	26,05	2840,16±8,57
		Opt3 64 bit	6713,46	43,00	6713,46±14,15
	WinKVM	No Opt	111,40	8,59	111,40±2,83
		Opt3 64 bit	275,54	42,58	275,54±14,01
Linpack	VMware	No Opt	359,22	1,23	359,22±0,40
		Opt3 64 bit	1655,13	15,90	1655,13±5,23
	Virtual Box	No Opt	349,61	1,91	349,61±0,63
		Opt3 64 bit	1616,18	8,71	1616,18±2,87
	WinKVM	No Opt	6,07	0,31	6,07±0,10
		Opt3 64 bit	9,62	0,33	9,62±0,11

Następnym wykorzystanym benchmarkiem był Linpack. Na rys. 7 oraz w tabeli 1 przedstawiono zestawienie otrzymanych wyników dla tego testu. Odchylenia standardowe są niewielkie i wyraźnie widać, że najwyższa wydajność w tym teście została uzyskana dla hypervisora VMware. Ponownie, implementacja KVM w systemie Windows jest zdecydowanie gorsza niż pozostałe, testowane rozwiązania.

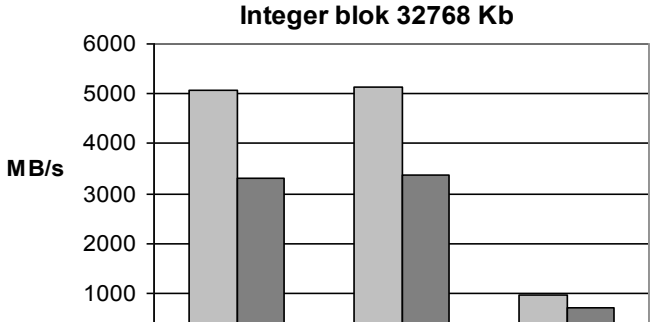
Ostatnim przeprowadzonym testem był test pamięci RAMSMP. Wyniki pomiarów dla tego testu zostały przestawione na rys. 8 oraz w tabeli 2. Wskazują one na niewielką przewagę hypervisora Virtual Box nad VMware. Analizując dane zawarte na rys. 8 oraz w tabeli 2, można stwierdzić, iż proces odczytu i zapisu dużych bloków pamięci w klastrze Hadoop wykorzystującym maszyny wirtualne uruchomione w Virtual Box jest nieznacznie wydajniejszy w stosunku do klastra prywatnego wykorzystującego maszyny wirtualne uruchomione w VMware Player.

a)



	VMware	Virtual Box	WinKVM
Reading	9079,707	9447,671	1101,4576
Writing	6087,842	6009,465	768,9194

b)



	VMware	Virtual Box	WinKVM
Reading	5059,8076	5140,7416	951,1666
Writing	3311,9802	3353,988	692,8752

Rys. 8. Wyniki testu RAMSMP
Fig. 8. Results of RAMSMP benchmark

Tabela 2
Zestawienie odchyleń standardowych oraz przedziałów ufności dla testu RAMSMP

	VMware		Virtual Box		WinKVM	
Test	Odch. st.	Przedział ufności	Odch. st.	Przedział ufności	Odch. st.	Przedział ufności
1024 Kb odczyt	483,12	9079,71±158,94	573,66	9447,67±188,73	13,85	1101,46±4,6
1024 Kb zapis	446,62	6087,84±146,93	496,25	6009,47±163,26	7,22	768,92±2,37
32768 Kb odczyt	29,37	5059,81±9,66	91,53	5140,74±30,11	11,36	951,17±3,73
32768 Kb zapis	12,53	3311,98±4,12	20,23	3353,99±6,65	5,01	692,88±1,64

4. Podsumowanie

W artykule dokonano porównania wpływu zarządcy maszyny wirtualnej na wydajność systemów eksploracji danych na przykładzie dystrybucji Cloudera Hadoop. Porównano wpływ hipervisora maszyny wirtualnej na typowe operacje wykonywane w testowym, prywatnym klastrze Hadoop. Klaster ten wykorzystywał standardowe komputery PC, systemem operacyjnym gospodarza był MS Windows 7 x64, a systemem gościa stanowiła dystrybucja Cloudera Hadoop x64. Na podstawie otrzymanych wyników można stwierdzić, iż hypervisory VMware oraz Virtual Box oferują czas wykonania zadań testowych w klastrze Hadoop na zbliżonym poziomie, natomiast implementacja WinKVM jest ponad 10-krotnie wolniejsza. Aby uchwycić różnice we wpływie hypervisorów, zostały przeprowadzone kolejne testy wydajnościowe. Ich ciekawym aspektem jest ocena porównawcza narzutu systemu operacyjnego gościa uruchamianego wirtualnie. Dla hypervisora VMware okazał się on najmniejszy i wyniósł około 20%, a dla pozostałych dwóch rozwiązań jego wartość wynosiła powyżej 30%. W związku z tym można by się było spodziewać lepszej wydajności klastra Hadoop uruchamianego pod kontrolą zarządcy VMware. Hipoteza ta jednak nie potwierdziła się, a kolejne testy wskazały na obszary, w których wpływ hypervisorów maszyn wirtualnych jest bardzo widoczny. Klaster Hadoop uruchomiony pod kontrolą zarządcy VMware cechuje się większą wydajnością obliczeń obciążających CPU, natomiast zarządca Virtual Box jest nieznacznie lepszy przy obliczeniach wymagających operacji na blokach pamięci operacyjnej o dużym rozmiarze. Z tego też powodu wytrenowanie klasyfikatora dla przykładowych danych z użyciem algorytmu regresji logistycznej dało podobne rezultaty przy hypervisorach VMware i Virtual Box.

Podsumowując, przeprowadzone badania wykazały, iż istnieje wpływ zarządcy maszyny wirtualnej na wydajność systemów eksploracji badanych. Wielkość i charakter tego wpływu

zależy od rodzaju zadań uruchamianych w klastrze Hadoop. Zadania wymagające operacji na dużych blokach pamięci są najlepiej obsługiwane przez hypervisora Virtual Box, natomiast zadania wymagające większej wydajności CPU przez hypervisora VMware. Nie można zatem wskazać jednoznacznie rozwiązania, które jest najlepsze.

BIBLIOGRAFIA

1. Yao Ke-Thia, Lucas R., Gottschalk T., Wagenbreth G., Ward C.: Data Analysis for Massively Distributed Simulations, Interservice/Industry Training, Simulation, and Education Conference IITSEC 2009 Paper No. 9350.
2. Dean J., Sanjay Ghemawat S.: MapReduce: Simplified Data Processing on Large Clusters, http://research.google.com/archive/mapreduce.html, March 2012.
3. Hadoop Blog, http://developer.yahoo.com/blogs/hadoop/, March 2012.
4. Welcome to Hadoop Apache, http://hadoop.apache.org, March 2012.
5. Lam C.: Hadoop in Action, Manning Publications Co. 2011.
6. Nurmi D., Wolski R., Grzegorczyk Ch., Obertelli G., Soman S., Youseff L., Zagorodnov D.: The Eucalyptus Open-source Cloud-computing System, 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID), Vol. 0, 2009, s. 124÷131.
7. CDH Version and Packaging Information – Cloudera Support, https://ccp.cloudera.com-/display/DOC/CDH+Version+and+Packaging+Information, March 2012.

Wpłynęło do Redakcji 18 marca 2012 r.

Abstract

Data mining systems play a very important role in the IT industry. They are used in many fields of science as well as apply in the commercial solutions. Hadoop project is a data mining system developed by the Apache. It is part of the Cloudera distribution and includes a set of libraries for creating tasks using Map Reduce paradigm. Hadoop is used among others entities such as: Facebook, New York Times, Samsung, Google, Yahoo, Amazon. That is why the paper presents a comparative analysis of the impact of the virtual machine manager on the data mining systems. Discussion is based on the results obtained in a test environment based on the Cloudera Hadoop distribution. Tested virtualization hypervisors were: VMware, Virtual Box and WinKVM. Four different tests were planed and conducted in order to asses

impact of the hypervisor on data mining system efficiency. They were: classifier training with logistic regression algorithm, Dhrystones, Linpack and RAMSMP benchmarks. The results of the carried out tests show that there is no distinct winner, in some areas the VMware hypervisor performs better (CPU), but in the other the Virtual Box hypervisor points out (RAM).

Adres

Dariusz CZERWIŃSKI: Politechnika Lubelska, Instytut Podstaw Elektrotechniki i Elektrotechnologii, ul. Nadbystrzycka 38a, 20-618 Lublin, Polska, d.czerwinski@pollub.pl