

Mateusz SMOLIŃSKI  
Politechnika Łódzka, Instytut Informatyki

## WPŁYW MIGAWEK PRZESTRZENI SKŁADOWANIA NA WYDAJNOŚĆ PRZETWARZANIA TRANSAKCJI W RELACYJNEJ BAZIE DANYCH

**Streszczenie.** W artykule przedstawiono wpływ migawek przestrzeni składowania na wydajność przetwarzania transakcji w relacyjnych bazach danych. Przedstawiono wyniki badań wydajności przetwarzania transakcji dla różnych typów migawek wykonanych dla jedno- i wielodyskowych przestrzeni składowania relacyjnych baz danych, a zarządzane przez PostgreSQL i MariaDB. Wyniki badań uzyskane dla różnych przestrzeni składowania relacyjnych baz danych pozwoliły ustalić, która konfiguracja zapewnia obsługę migawek jednocześnie minimalizując spadek wydajności przetwarzania transakcji bazodanowych.

**Słowa kluczowe:** wydajność przetwarzania transakcji, migawki przestrzeni składowania relacyjnej bazy danych

## INFLUENCE OF THE STORAGE SPACE SNAPSHOTS ON TRANSACTION PROCESSING IN RELATIONAL DATABASE

**Summary.** This paper presents influence of storage snapshots in various relational database storage space configuration on performance in database transaction processing. The obtained results for single and multi disk storage space configurations in popular database management systems like PostgreSQL and MariaDB allow to choose storage space configuration for relational database, which support snapshotting and minimize performance drop in database transaction processing.

**Keywords:** transaction processing performance, snapshots in relational database storage space

## 1. Przetwarzanie transakcji w relacyjnych bazach danych

Systemy informatyczne zaliczane do klasy OLTP (*Online Transaction Processing*) wymagają zastosowania mechanizmów zapewniających spójność, kompletność i integralność przetwarzanych danych. Mechanizm transakcji spełniający wymagania ACID (*Atomicity, Consistency, Isolation, Durability*) umożliwia ochronę równoległe przetwarzanych danych [1, 8]. Zastosowanie mechanizmu transakcji wymaga oprogramowania, zapewniającego realizację operacji na danych w ramach przetwarzanych transakcji. Przykładem takiego oprogramowania jest system zarządzania bazami danych DBMS (*Database Management System*), obsługujący bazy danych. Systemy informatyczne często przechowują dane w relacyjnych bazach danych, które dodatkowo zapewniają zachowanie zależności pomiędzy przechowywanymi w tabelach rekordami. Popularność zastosowania relacyjnego modelu danych do przechowywania danych w bazach przez systemy informatyczne wynika także ze standaryzacji struktur danych oraz wykonywanych operacji, zastosowania typizacji, możliwości definiowania ograniczeń definiujących poprawność przechowywanych danych oraz mechanizmów usprawniających dostęp do danych przechowywanych w bazie.

Wydajność systemów OLTP może być mierzona jako liczba identycznych transakcji, które zostały zatwierdzone w ustalonym czasie. Standaryzacja operacji wykonywanych w granicach transakcji, struktur danych i czasu pomiaru pozwala porównywać wydajność różnych systemów [3]. Jeżeli w systemie informatycznym dane są przetwarzane w ramach lokalnych transakcji bazodanowych, to konfiguracja przestrzeni składowania relacyjnej bazy danych wpływa na wydajność przetwarzania danych. Konfiguracja przestrzeni składowania relacyjnych bazy danych w większości DBMS obejmuje system plików wraz z lokalnym urządzeniem blokowym, w którym system plików został utworzony [2, 6]. Zapewnienie wyższej wydajności w realizacji operacji odczytu i zapisu bloku w urządzeniu wymaga zastosowania wielu fizycznych urządzeń blokowych (np. dysków). Zarządzenie wielodyskową przestrzenią pamięci masowej wymaga zastosowania menadżera urządzeń blokowych, który może być implementowany sprzętowo lub programowo. Niezależnie od zastosowanego menadżera każdy wolumen logiczny wymaga określenia liczby dysków oraz polityki alokacji, która ustala zasady adresacji i lokalizacji bloków na blokowych urządzeniach fizycznych. Współczesne systemy operacyjne, np. GNU/Linux, udostępniają różne implementacje menadżerów urządzeń blokowych, które mogą być dostarczane przez nowoczesne systemy plików lub są implementowane jako logiczne urządzenia blokowe, m.in. LVM (*Logical Volume Manager*) [7].

W badaniach wykorzystano dwie popularne implementacje DBMS: PostgreSQL oraz MariaDB. Przed każdym pomiarem wydajności lokalnie przetwarzanych transakcji bazodano-

wych wykonano konfigurację przestrzeni składowania danych relacyjnej bazy oraz utworzono migawki. Zastosowanie migawek przestrzeni składowania pozwala zachować stan przestrzeni składowania z momentu utworzenia migawki. Utworzenie migawki umożliwia więc zachowanie stanu relacyjnych baz danych obsługiwanych przez DBMS w ustalonych chwilach czasu. Jednak utworzona migawka wprowadza dodatkowe operacje wejścia/wyjścia, w wyniku których wydajność przetwarzania transakcji może zostać obniżona. Przeprowadzone badania wydajności przetwarzania transakcji bazodanowych uwiarygodniły, które konfiguracje przestrzeni składowania danych i dla jakiego typu migawek zapewniają najmniejszy spadek wydajności. Informacje te umożliwiają skonfigurowanie wydajnej i elastycznej przestrzeni składowania danych dla relacyjnych baz zlokalizowanych w wymienionych systemach DBMS.

## 2. Środowisko do pomiaru przetwarzanych transakcji

Porównywanie konfiguracji przestrzeni składowania relacyjnych baz danych wymaga zastosowania identycznego środowiska dla przetwarzania transakcji bazodanowych, metody pomiarowej i zawartości relacyjnej bazy danych. Jednolite środowisko, w którym przeprowadzono pomiary wydajności przetwarzania transakcji w relacyjnej bazie danych, składa się zarówno z zasobów sprzętowych, jak i programowych. Do pomiarów wykorzystano 64-bitowy komputer wyposażony w: procesor Intel i5-2400 CPU 3.10 GHz, 8 GB pamięci RAM oraz pięć identycznych dysków SATA-3 Western Digital model WD5000AZRX o pojemności 466 GB i 64 MB pamięci cache. W zewnętrznej pamięci masowej podłączonej poprzez USB zainstalowano system operacyjny Linux z dystrybucji Fedora 21, po wykonaniu aktualizacji oprogramowania zastosowano jądro systemu w wersji 3.17.8-300.fc21.x86\_64 [5]. W konfiguracjach zachowano domyślne konwencje konfiguracji.

W badaniach tworzone konfiguracje lokalnej przestrzeni składowania danych z zastosowaniem następujących systemów plików: BTRFS (btrfs-progs-3.18.1-1.fc21.x86\_64), EXT4 (e2fsprogs-1.42.11-4.fc21.x86\_64), FAT32 (dosfstools-3.0.27-1.fc21.x86\_64), JFS (jfsutils-1.1.15-5.fc21.x86\_64), XFS (xfsprogs-3.2.1-2.fc21.x86\_64). Systemy plików były tworzone w wolumenach logicznych zarządzanych przez menadżera LVM (lvm2-2.02.111-1.fc21.x86\_64), który udostępnia różne polityki alokacji na wielu urządzeniach blokowych. W każdej konfiguracji z zastosowaniem LVM wykorzystywano domyślny rozmiar jednostki alokacji *extent* 4 MB. Wszystkie pomiary liczby przetwarzanych transakcji w relacyjnej bazie danych zostały wykonane przez narzędzie *sysbench* w wersji 0.4.12, które współpracuje

z różnymi DBMS, m.in. MariaDB oraz PostgreSQL. W badaniach zastosowano oprogramowanie MariaDB w wersji 10.0.15 z silnikiem InnoDB oraz PostgreSQL w wersji 9.3.5.

### 3. Scenariusze testowania wydajności przetwarzania transakcji w bazie

Scenariusz testowania wydajności w przetwarzaniu transakcji bazodanowych uwzględnia konfigurację środowiska, w szczególności przestrzeni składowania danych, gdzie zlokalizowane były relacyjne bazy danych. W scenariuszach testowych wyróżniono dwa rodzaje konfiguracji – z zastosowaniem zewnętrznego menadżera LVM lub stosując wewnętrzny menadżer dysków systemu plików BTRFS. Każdy z wymienionych menadżerów dysków obsługuje różne polityki alokacji danych, które określają zasady dystrybucji danych pomiędzy wieloma dyskami, np. polityki alokacji macierzowej RAID (Redundant Array of Inexpensive Disks) [4]. Zewnętrzny menadżer wolumenów oprócz zwykłych wolumenów umożliwia też tworzenie cienkich wolumenów z zastosowaniem opóźnionej alokacji bloków TP (Thin Provisioning). Zarówno w zwykłym, jak i cienkim wolumenie logicznym utworzonym z zastosowaniem LVM można umieścić dowolny system plików. W badaniach ograniczono zbiór systemów plików do popularnych formatów: BTRFS, EXT4, FAT32, JFS, XFS [4, 5, 6].

System plików BTRFS oprócz wewnętrznego menadżera wolumenów udostępnia możliwość wydzielenia podwolumenów (*subvolumes*), dla których można tworzyć migawki. Menadżer LVM także oferuje możliwość zachowania stanu dla wolumenu logicznego przez utworzenie migawki. Należy zauważyć, że w przypadku cienkich wolumenów tworzenie migawki jest realizowane inaczej niż dla zwykłego wolumenu logicznego LVM. Utworzona migawka cienkiego wolumenu ma własny cykl życia, który nie jest zależny od dalszego istnienia cienkiego wolumenu, dla którego utworzono migawkę.

Scenariusz testowania obejmuje konfigurację logicznego urządzenia blokowego oraz systemu plików. Podstawowymi parametrami konfiguracyjnymi logicznego urządzenia blokowego jest zastosowany menadżer dysków, liczba wykorzystanych dysków i algorytm alokacji. W przypadku konfiguracji systemu plików istotny jest jego typ oraz rozmiar bloku. W celu ujednolicenia rozmiar bloku systemu plików został ustalony na 4 KB. W scenariuszach z zastosowaniem wewnętrznego menadżera dysków BTRFS nie można stosować innego typu systemu plików.

W każdym scenariuszu testowym skonfigurowaną przestrzeń dyskową udostępniono poprzez montowanie do przechowywania klastra baz danych DBMS (jako domyślną przestrzeń tabel), odpowiednio dla MariaDB i PostgreSQL. W każdym DBMS tworzono identyczną relacyjną bazę danych, zawierającą 10 milionów rekordów. Rozmiar utworzonej bazy przekra-

czał 2 GB. Zarówno struktury, jak i przechowane dane relacyjnej bazy zostały wygenerowane z wykorzystaniem narzędzia *sysbench*. Przed dokonaniem pomiaru wydajności przetwarzania transakcji bazodanowych tworzone były wszystkie migawki, które były wymagane w scenariuszu konfiguracji przestrzeni składowania relacyjnych baz danych. Pomiar liczby przetwarzanych transakcji w jednostce czasu został zrealizowany poprzez test OLTP udostępniany przez narzędzie *sysbench*. W trakcie testu program *sysbench* uruchamia wątki generujące sekwencję transakcji bazodanowych, każda z transakcji zawiera operacje wymagające niesekwencyjnego dostępu do przechowywanych w bazie danych. W przeciwieństwie do komercyjnych testów przetwarzania transakcji, m.in. TPC (*Transaction Processing Performance Council*), narzędzie *sysbench* nie dysponuje różnymi modelami testów przetwarzania transakcji. W celu uniknięcia sytuacji wzajemnego zablokowania (*deadlock*) wątków powołanych przez program *sysbench* zawsze uruchamiany był tylko jeden wątek. Po zakończeniu przetwarzania transakcji program *sysbench* udostępnia statystyki, wśród których prezentowana jest uśredniona liczba zatwierdzonych transakcji w trakcie sekundy (TX/s).

#### **4. Analiza wydajności przetwarzania transakcji w scenariuszach testowych**

W podrozdziałach zaprezentowano analizy dla różnych zestawień scenariuszy konfiguracji przestrzeni składowania relacyjnych baz danych. Zaprezentowane analizy w szczególności uwzględniają wpływ różnych typów migawek przestrzeni składowania relacyjnych baz danych na wydajność przetwarzania transakcji bazodanowych.

##### **4.1. Wpływ migawek LVM na wydajność przetwarzania transakcji bazodanowych**

Porównanie scenariuszy testowych, gdzie dystrybucja bloków pomiędzy dyskami była realizowana przez menadżera LVM zgodnie z algorytmem RAID poziomego zero (ozn. RAID0), pozwala określić dla wykorzystanych w badaniach DBMS, który typ systemu plików ma struktury minimalizujące wpływ liczby utworzonych migawek LVM na wydajność przetwarzanych transakcji bazodanowych.

W tabeli 1 zaprezentowano wyniki pomiaru wydajności przetwarzania transakcji bazodanowych TX/s (uśredniona liczba zatwierdzonych transakcji w trakcie jednej sekundy) dla przestrzeni składowania złożonej z systemu plików BTRFS zlokalizowanego w wolumenie logicznym zarządzanym przez LVM z polityką alokacji RAID0 o określonej liczbie dysków z zakresu od 1 do 5. Przedstawione wyniki uwiadcniają wpływ liczby migawek utworzonych dla zwykłego wolumenu LVM na wydajność przetwarzania transakcji bazodanowych w

jedno- i wielodyskowych konfiguracjach przestrzeni składowania relacyjnych baz danych. Taki sam zbiór pomiarów wydajności przetwarzania transakcji bazodanowych dla identycznego zbioru scenariuszy konfiguracji przestrzeni składowania relacyjnych baz danych, gdzie zastosowano inny system plików: EXT4, FAT32, JFS i XFS, przedstawiają odpowiednio kolejne tabele 2-5.

Tabela 1

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików BTRFS zlokalizowanym w wolumenie LVM z polityką alokacji RAID0

Liczba migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
BTRFS zlokalizowany w wolumenie LVM (1 dysk)	11,07	4,19	2,80	2,03	1,17	MariaDB
	9,28	5,13	4,46	3,38	2,44	PostgreSQL
BTRFS zlokalizowany w wolumenie LVM (2 dyski)	13,7	6,09	4,17	2,37	1,3	MariaDB
	11,49	6,38	4,96	3,56	2,41	PostgreSQL
BTRFS zlokalizowany w wolumenie LVM (3 dyski)	16,65	7,68	5,45	2,84	1,49	MariaDB
	14,49	7,69	6,09	4,06	2,63	PostgreSQL
BTRFS zlokalizowany w wolumenie LVM (4 dyski)	17,43	8,59	5,60	2,60	1,52	MariaDB
	15,24	8,51	6,15	4,01	2,54	PostgreSQL
BTRFS zlokalizowany w wolumenie LVM (5 dysków)	19,66	9,33	6,35	3,09	1,45	MariaDB
	17,92	8,92	6,33	4,67	2,53	PostgreSQL

Tabela 2

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików EXT4 zlokalizowanym w wolumenie LVM z polityką alokacji RAID0

Liczba migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
EXT4 zlokalizowany w wolumenie LVM (1 dysk)	11,58	2,51	2,13	0,96	0,66	MariaDB
	14,14	6,47	4,28	2,31	0,80	PostgreSQL
EXT4 zlokalizowany w wolumenie LVM (2 dyski)	14,66	5,42	3,57	1,92	0,96	MariaDB
	18,06	8,85	7,23	4,97	3,11	PostgreSQL
EXT4 zlokalizowany w wolumenie LVM (3 dyski)	16,06	6,39	4,11	1,96	1,06	MariaDB
	20,56	10,55	8,08	5,35	3,23	PostgreSQL
EXT4 zlokalizowany w wolumenie LVM (4 dyski)	17,54	6,39	4,13	1,92	1,08	MariaDB
	23,37	12,00	8,80	5,11	3,03	PostgreSQL
EXT4 zlokalizowany w wolumenie LVM (5 dysków)	18,37	6,67	3,81	1,95	1,04	MariaDB
	25,39	13,14	9,46	6,14	3,42	PostgreSQL

Analiza danych zaprezentowanych w tabelach 1 i 2 wykazuje, że system plików BTRFS zlokalizowany w wolumenie LVM bez migawek zapewnia podobny jak w przypadku EXT4 poziom wydajności w przetwarzaniu transakcji w relacyjnej bazie danych obsługiwanej przez MariaDB. Jednak system plików EXT4 wykazuje w tym przypadku mniejszą wydajność w sytuacji, gdy zostaną utworzone migawki. Każdy scenariusz konfiguracji przestrzeni składowania danych z EXT4, w którym relacyjna baza zlokalizowana jest w PostgreSQL, wyka-

zuje wyższą wydajność przetwarzania transakcji bazodanowych niż w przypadku zastosowania BTRFS.

Tabela 3

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików FAT32 zlokalizowanym w wolumenie LVM z polityką alokacji RAID0

Liczba migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
FAT32 zlokalizowany w wolumenie LVM (1 dysk)	12,94	9,01	8,83	6,49	4,6	MariaDB
	10,13	5,5	4,68	3,68	2,68	PostgreSQL
FAT32 zlokalizowany w wolumenie LVM (2 dyski)	15,63	12,52	11,28	8,34	5,66	MariaDB
	13,05	7,47	6,35	4,67	3,04	PostgreSQL
FAT32 zlokalizowany w wolumenie LVM (3 dyski)	17,22	15,33	13,52	9,59	6,22	MariaDB
	15,39	8,91	7,22	5,07	3,21	PostgreSQL
FAT32 zlokalizowany w wolumenie LVM (4 dyski)	17,98	15,11	12,77	8,77	5,88	MariaDB
	16,83	9,92	7,34	4,83	3,01	PostgreSQL
FAT32 zlokalizowany w wolumenie LVM (5 dysków)	19,06	15,51	13,08	9,85	5,85	MariaDB
	18,55	9,94	7,22	5,32	3,06	PostgreSQL

Tabela 4

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików JFS zlokalizowanym w wolumenie LVM z polityką alokacji RAID0

Liczba migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
JFS zlokalizowany w wolumenie LVM (1 dysk)	20,36	14,03	11,84	8,44	5,16	MariaDB
	15,36	6,69	5,63	4,14	2,74	PostgreSQL
JFS zlokalizowany w wolumenie LVM (2 dyski)	23,21	16,08	14,36	10,14	6,13	MariaDB
	21,48	9,8	7,64	5,08	3,25	PostgreSQL
JFS zlokalizowany w wolumenie LVM (3 dyski)	28,39	19,44	16,99	10,49	6,77	MariaDB
	26,37	11,54	9,26	5,75	3,41	PostgreSQL
JFS zlokalizowany w wolumenie LVM (4 dyski)	31,71	20,34	16,54	11,28	6,10	MariaDB
	32,9	20,02	16,4	11,47	6,5	PostgreSQL
JFS zlokalizowany w wolumenie LVM (5 dysków)	32,9	20,02	16,4	11,47	6,5	MariaDB
	36,97	13,71	9,58	5,96	3,12	PostgreSQL

Dalsza analiza wyników wykazała, że spośród wszystkich wykorzystanych systemów plików zlokalizowanych w wolumenie LVM najwyższą wydajność w przetwarzaniu transakcji bazodanowych zapewnia przestrzeń składowania baz danych utworzona z zastosowaniem systemu plików JFS. We wszystkich przypadkach, oprócz scenariuszy z systemu plików EXT4 zlokalizowanego w wolumenie LVM, widoczna jest wyższa wydajność przetwarzania transakcji w relacyjnej bazie danych obsługiwanej przez MariaDB niż przez PostgreSQL.

Dla każdego typu systemu plików, który był zlokalizowany w zwykłym wolumenie LVM tworząc przestrzeń składowania dla relacyjnych baz danych, można zaobserwować zależność wzrostu liczby przetwarzanych transakcji wraz ze wzrostem liczby dysków. Liczba zatwier-

dzonych transakcji zmniejsza się wraz ze zwiększaniem liczby migawek dla zwykłego wolumenu LVM.

Tabela 5

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików XFS zlokalizowanym w wolumenie LVM z polityką alokacji RAID0

Liczba migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
XFS zlokalizowany w wolumenie LVM (1 dysk)	13,83	6,22	4,1	2,6	1,46	MariaDB
	13,95	6,69	5,57	4,28	2,88	PostgreSQL
XFS zlokalizowany w wolumenie LVM (2 dyski)	16,58	8,04	6,08	3,53	1,91	MariaDB
	17,56	9,02	7,3	5,09	3,24	PostgreSQL
XFS zlokalizowany w wolumenie LVM (3 dyski)	17,86	10,14	7,6	3,83	1,98	MariaDB
	20,58	10,88	8,73	5,62	3,37	PostgreSQL
XFS zlokalizowany w wolumenie LVM (4 dyski)	19,04	11,27	7,41	3,82	1,84	MariaDB
	23,28	12,30	9,04	5,31	3,15	PostgreSQL
XFS zlokalizowany w wolumenie LVM (5 dysków)	21,7	12,16	7,33	4,47	2,04	MariaDB
	25,75	13,37	9,91	6,26	3,45	PostgreSQL

#### 4.2. Wpływ cienkich migawek LVM na wydajność przetwarzania transakcji bazodanowych

Tabela 6

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików BTRFS zlokalizowanym w cienkim wolumenie LVM z polityką alokacji puli RAID0

Liczba cienkich migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
BTRFS zlokalizowany w cienkim wolumenie LVM (1 dysk)	7,69	4,52	3,58	2,11	1,35	MariaDB
	6,93	5,00	4,27	3,27	2,29	PostgreSQL
BTRFS zlokalizowany w cienkim wolumenie LVM (2 dyski)	13,7	14,16	13,13	12,9	13,29	MariaDB
	11,49	10,35	9,97	10,58	10,04	PostgreSQL
BTRFS zlokalizowany w cienkim wolumenie LVM (3 dyski)	14,73	14,79	14,75	14,34	13,54	MariaDB
	11,14	11,08	11,18	10,79	11,5	PostgreSQL
BTRFS zlokalizowany w cienkim wolumenie LVM (4 dyski)	16,3	15,4	16,46	14,28	15,41	MariaDB
	12,55	12,1	11,7	12,66	12,29	PostgreSQL
BTRFS zlokalizowany w cienkim wolumenie LVM (5 dysków)	14,98	15,65	15,46	13,23	13,29	MariaDB
	12,32	12,56	11,06	11,26	10,58	PostgreSQL

Cienkie wolumeny LVM opóźniają alokację bloku do momentu pierwszej operacji wejścia/wyjścia, która wymaga dostępu do przechowywanych w bloku danych. Bloki alokowane są z puli zasobów, która w zależności od scenariusza została zlokalizowana na jednym lub kilku dyskach (zastosowano algorytm alokacji RAID poziomemu 0 dla puli). Zatem rozmiar cienkiego wolumenu nie musi być zgodny z łącznym rozmiarem przydzielonych mu obszarów w fizycznych urządzeniach blokowych. W odróżnieniu od zwykłych migawek LVM każda migawka utworzona dla cienkiego wolumenu LVM nie posiada statycznego limitu



rozmiaru swoich buforów, ponieważ jest niezależnym cienkim wolumenem. Taka migawka może istnieć nawet po usunięciu cienkiego wolumenu, dla którego została stworzona.

W tabelach 6-10 zaprezentowano zestawienia scenariuszy konfiguracji przestrzeni składowania relacyjnych baz danych dla różnych systemów plików zlokalizowanych w cienkich wolumenach LVM, z uwzględnieniem liczby utworzonych cienkich migawek LVM. Dla ustalonego typu systemu plików analiza wyników uzyskanych w scenariuszach z wykorzystaniem zwykłych i cienkich wolumenów LVM wykazała niewielki spadek wydajności w przetwarzaniu transakcji, gdy nie utworzono żadnych wolumenów migawek.

Tabela 7

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików EXT4 zlokalizowanym w cienkim wolumenie LVM z polityką alokacji puli RAID0

Liczba cienkich migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
EXT4 zlokalizowany w cienkim wolumenie LVM (1 dysk)	10,1	2,47	1,42	0,56	0,15	MariaDB
	12,51	5,29	3,61	1,96	0,97	PostgreSQL
EXT4 zlokalizowany w cienkim wolumenie LVM (2 dyski)	14,66	14,34	14,29	14,57	14,56	MariaDB
	18,06	13,01	13,61	13,70	13,93	PostgreSQL
EXT4 zlokalizowany w cienkim wolumenie LVM (3 dyski)	16,18	15,88	15,4	15,72	15,65	MariaDB
	19,11	15,21	14,69	15,23	15,08	PostgreSQL
EXT4 zlokalizowany w cienkim wolumenie LVM (4 dyski)	16,86	16,89	16,84	17,12	17,15	MariaDB
	21,06	16,86	16,57	16,79	16,99	PostgreSQL
EXT4 zlokalizowany w cienkim wolumenie LVM (5 dysków)	18,01	15,58	15,59	15,81	15,82	MariaDB
	23,58	15,8	16,97	15,94	17,36	PostgreSQL

Tabela 8

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików FAT32 zlokalizowanym w cienkim wolumenie LVM z polityką alokacji puli RAID0

Liczba cienkich migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
FAT32 zlokalizowany w cienkim wolumenie LVM (1 dysk)	12,36	8,93	8,25	6,32	3,91	MariaDB
	9,71	5,63	4,96	3,41	2,49	PostgreSQL
FAT32 zlokalizowany w cienkim wolumenie LVM (2 dyski)	15,63	14,66	14,72	14,82	14,54	MariaDB
	13,05	10,01	10,33	10,30	10,33	PostgreSQL
FAT32 zlokalizowany w cienkim wolumenie LVM (3 dyski)	15,96	16,69	16,69	16,56	16,53	MariaDB
	13,79	11,56	11,61	11,57	11,91	PostgreSQL
FAT32 zlokalizowany w cienkim wolumenie LVM (4 dyski)	17,22	17,55	17,78	17,41	17,36	MariaDB
	14,84	13,34	13,31	12,74	12,81	PostgreSQL
FAT32 zlokalizowany w cienkim wolumenie LVM (5 dysków)	17,95	17,01	16,82	16,52	16,34	MariaDB
	16,34	12,51	13,07	12,56	12,24	PostgreSQL

Analiza wydajności przetwarzania transakcji, gdy przestrzeń składowania baz danych jest skonfigurowana z zastosowaniem cienkich wolumenów, dla których utworzono migawki, uwidacznia znaczną przewagę w liczbie zatwierdzonych transakcji w stosunku do zwykłych

wolumenów z migawkami. Zależność ta jest widoczna dla każdego z uczestniczących w badaniach systemów plików, w szczególności porównując podobne scenariusze, gdzie utworzono więcej niż 2 migawki.

Tabela 9

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików JFS zlokalizowanym w cienkim wolumenie LVM z polityką alokacji puli RAID0

Liczba cienkich migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
JFS zlokalizowany w cienkim wolumenie LVM (1 dysk)	19,88	13,61	12,61	7,84	5,09	MariaDB
	14,08	7,30	5,97	3,66	2,43	PostgreSQL
JFS zlokalizowany w cienkim wolumenie LVM (2 dyski)	23,21	23,83	23,63	23,56	23,3	MariaDB
	21,48	16,86	16,88	17,05	17,10	PostgreSQL
JFS zlokalizowany w cienkim wolumenie LVM (3 dyski)	25,37	27,83	26,93	26,95	27,17	MariaDB
	22,25	20,74	20,75	21,01	20,53	PostgreSQL
JFS zlokalizowany w cienkim wolumenie LVM (4 dyski)	28,19	30,18	30,21	30,28	30,02	MariaDB
	26,35	24,28	23,83	24,52	23,38	PostgreSQL
JFS zlokalizowany w cienkim wolumenie LVM (5 dysków)	30,75	27,76	27,27	28,15	28,16	MariaDB
	30,44	24,58	25,07	25,23	25,61	PostgreSQL

Tabela 10

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików XFS zlokalizowanym w cienkim wolumenie LVM z polityką alokacji puli RAID0

Liczba cienkich migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
XFS zlokalizowany w cienkim wolumenie LVM (1 dysk)	14,51	6,23	4,55	2,53	1,35	MariaDB
	14,01	6,32	5,81	3,85	2,63	PostgreSQL
XFS zlokalizowany w cienkim wolumenie LVM (2 dyski)	16,58	16,34	16,31	16,42	16,4	MariaDB
	17,56	13,34	13,66	13,74	14,21	PostgreSQL
XFS zlokalizowany w cienkim wolumenie LVM (3 dyski)	19,37	17,69	17,6	17,39	17,24	MariaDB
	19,34	16,48	15,93	15,06	16,48	PostgreSQL
XFS zlokalizowany w cienkim wolumenie LVM (4 dyski)	20,18	18,91	18,25	18,30	18,36	MariaDB
	21,16	17,97	16,32	16,39	17,82	PostgreSQL
XFS zlokalizowany w cienkim wolumenie LVM (5 dysków)	21,76	19,3	18,99	18,51	18,51	MariaDB
	23,35	18,58	16,74	17,16	17,78	PostgreSQL

#### 4.3. Wpływ migawek BTRFS na wydajność przetwarzania transakcji bazodanowych

W zestawieniu wyników przedstawionym w tabeli 11 zaprezentowano scenariusze konfiguracji, w których przestrzeń składowania relacyjnych baz danych została zlokalizowana w podwolumenie systemu plików BTRFS. W zestawieniu przedstawiono konfiguracje BTRFS zlokalizowane w jednym i wielu dyskach, które są zarządzane przez wbudowanego

menadżera dysków BTRFS. Zestawienie przedstawia wpływ migawek podwolumenów BTRFS na wydajność przetwarzania transakcji bazodanowych.

Tabela 11

Wydajność przetwarzania transakcji bazodanowych (TX/s) w scenariuszach z systemem plików BTRFS z wewnętrznym menadżerem dysków z polityką alokacji danych RAID0

Liczba cienkich migawek:	0	1	2	5	10	DBMS
Konf. przestrzeni składowania:						
Podwolumen BTRFS zlokalizowany w 1 dysku	13,14	11,57	11,30	11,26	11,30	MariaDB
	9,8	8,06	8,35	8,5	8,43	PostgreSQL
Podwolumen BTRFS zlokalizowany w 2 dyskach	13,19	13,18	12,78	12,91	13	MariaDB
	10,38	10,03	10,05	10,16	10,23	PostgreSQL
Podwolumen BTRFS zlokalizowany w 3 dyskach	13,52	14,64	14,67	14,48	14,71	MariaDB
	11,30	11,06	11,21	11,05	11,26	PostgreSQL
Podwolumen BTRFS zlokalizowany w 4 dyskach	15,51	15,53	15,77	15,85	15,85	MariaDB
	13,38	12,48	12,22	12,28	12,33	PostgreSQL
Podwolumen BTRFS zlokalizowany w 5 dyskach	16,43	16,48	16,56	16,55	16,64	MariaDB
	13,2	13,05	12,99	13,07	13,03	PostgreSQL

Analiza uzyskanych wyników pomiarów wydajności przetwarzania transakcji bazodanowych z tabel 1, 6 i 11 uwidoczniła fakt, że zastosowanie systemu plików BTRFS w konfiguracji przestrzeni składowania relacyjnych baz danych zapewnia najwyższą wydajność, jeżeli tylko zarządzanie dyskami jest realizowane przez wewnętrznego menadżera dysków w BTRFS. W dodatku scenariusze z bezpośrednim zarządzaniem dyskami przez BTRFS zapewniają najniższy spadek wydajności w przetwarzaniu transakcji przy rosnącej liczbie migawek.

## 5. Podsumowanie

Zapewnienie efektywnego przetwarzania transakcji bazodanowych wymaga zastosowania wydajnej przestrzeni przechowywania baz danych zarówno w warstwie urządzeń blokowych, jak i systemu plików. W wykonanych konfiguracjach przestrzeni składowania relacyjnych baz danych przy ustalonej liczbie dysków, menadżerze dysków i polityce alokacji najwyższą wydajność zapewniał system plików JFS. Ze względu na zastosowanie reguły COW (*Copy On Write*) wykorzystanie migawek dla zwykłych wolumenów LVM nie jest zalecane, gdyż już nawet jedna utworzona taka migawka może obniżyć o ponad połowę wydajność przetwarzania transakcji bazodanowych. Utworzenie 5 migawek dla zwykłego wolumenu LVM może zmniejszyć 12-krotnie wydajność przetwarzania transakcji bazodanowych. Przy ustalonym typie systemu plików degradacja wydajności w przetwarzaniu transakcji bazodanowych przy tworzeniu kolejnych migawek jest mniejsza, jeżeli tylko zastosowano cienkie wolumeny

lub podwolumeny BTRFS. Uzyskane wyniki badań pozwoliły też stwierdzić, że dla ustalonego scenariusza konfiguracji przestrzeni składowania relacyjnych baz danych (oprócz EXT4 i JFS zlokalizowanego w zwykłym wolumenie LVM) system MariaDB zapewnia wyższą wydajność przetwarzania transakcji bazodanowych niż w przypadku PostgreSQL.

## BIBLIOGRAFIA

1. Bernstein P. A., Newcomer E.: Principles of Transaction Processing. 2<sup>nd</sup> Edition. Morgan Kaufmann, 2009.
2. Bryant R., Forester R., Hawkes J.: Filesystem Performance and Scalability in Linux 2.4.17 Usenix. Proceeding of the Freenix Track, 2002, s. 259÷274.
3. Llanos D. R.: TPCC-UVa: An Open-Source TPC-C Implementation for Global Performance Measurement of Computer Systems. Sigmod Record, Vol. 35, Issue 4, 2006, s. 6÷15.
4. Rodeh O., Bacik J., Mason C.: BTRFS: The Linux B-Tree Filesystem. ACM Transactions on Storage, Vol. 9, Issue 3, 2013, s. 1÷32.
5. Sobell M. G.: Fedora and RedHat Enterprise Linux. 6<sup>th</sup> Edition. Prentice Hall, 2011.
6. Tanenbaum A. S., Bos H.: Modern Operating Systems. 4<sup>th</sup> Edition. Prentice Hall, 2014.
7. Teigland D., Muelshagen H.: Volume Managers in Linux. Proceedings of the Usenix Conference, 2001, s. 185÷198.
8. Vossen G., Weikum G.: Transactional Information Systems. Academic Press, 2002.

## Abstract

Relational databases are often used in on-line transaction processing systems. The main parameter that allows to determine system performance is number of committed transactions in the specified time interval. Relational database storage space configuration includes filesystem and block device layer, that affect performance in database transaction processing. Some modern filesystems and logical block devices have snapshotting feature. This paper presents influence of storage space snapshots in various database storage space configuration on performance in database transaction processing. The results of database transaction processing performance are presented in tables 1-11 for various types of storage space snapshot. All result of relational database transaction processing performance were presented for popular relational database management systems like PostgreSQL and MariaDB with various

storage space configurations. The obtained results for single and multi disk storage space configurations allow to choose storage space configuration for relational database, which support snapshotting and minimize performance drop in database transaction processing.

**Adres**

Mateusz SMOLIŃSKI: Politechnika Łódzka, Instytut Informatyki, ul. Wólczańska 215, 90-924 Łódź, Polska, mateusz.smolinski@p.lodz.pl.