



Kompendium wiedzy dla administratorów



Microsoft®

SQL Server® 2014

Podręcznik administratora

Adam Jorgensen, Bradley Ball, Steven Wort,
Ross LoForte, Brian Knight

Helion . The logo symbol consists of a stylized white 'H' shape with a checkmark-like element on the right side, set against a red background.

Tytuł oryginału: Professional SQL Server® 2014 Administration

Tłumaczenie: Tomasz Walczak

ISBN: 978-83-283-0673-8

Copyright © 2014 by John Wiley & Sons, Inc., Indianapolis, Indiana

All Rights Reserved.

This translation published under license with the original publisher John Wiley & Sons, Inc.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise without either the prior written permission of the Publisher.

The Wrox Brand trade dress is a trademark of John Wiley & Sons, Inc. in the United States and/or other countries. Used by permission.

Wiley, Wrox, the Wrox logo, Programmer to Programmer, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. SQL Server is a registered trademark of Microsoft Corporation. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc., is not associated with any product or vendor mentioned in this book

Translation copyright © 2015 by Helion S.A.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Wydawnictwo HELION

ul. Kościuszki 1c, 44-100 GLIWICE

tel. 32 231 22 19, 32 230 98 63

e-mail: helion@helion.pl

WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:

<ftp://ftp.helion.pl/przyklady/sql4pa.zip>

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/sql4pa>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

| | |
|--|-----------|
| O autorach | 21 |
| O współpracownikach | 23 |
| O redaktorach technicznych | 25 |
| Podziękowania | 27 |
| Wprowadzenie | 29 |
| | |
| Rozdział 1. Architektura systemu SQL Server 2014 | 35 |
| Środowisko systemu SQL Server 2014 | 35 |
| Nowe ważne funkcje wersji 2014 | 36 |
| Administratorzy produkcyjnych baz danych | 36 |
| Administrator rozwojowych baz danych | 37 |
| Administratorzy i programiści odpowiedzialni za analizy biznesowe | 38 |
| Architektura systemu SQL Server | 39 |
| Pliki bazy danych i dziennik transakcji | 39 |
| SQL Server Native Client | 40 |
| Standardowe systemowe bazy danych | 41 |
| Schematy | 43 |
| Synonimy | 44 |
| Obiekty DMO | 44 |
| Typy danych w systemie SQL Server 2014 | 45 |
| Edycje systemu SQL Server | 52 |
| Przegląd edycji | 52 |
| Licencje | 55 |
| Podsumowanie | 56 |
| | |
| Rozdział 2. Najlepsze praktyki związane z instalowaniem systemu SQL Server 2014 | 57 |
| Planowanie systemu | 58 |
| Opcje sprzętowe | 58 |
| Możliwości w obszarze oprogramowania i instalacji | 63 |

| | |
|---|----|
| Instalowanie systemu SQL Server | 66 |
| Nowe instalacje | 66 |
| Instalacje równoległe | 66 |
| Aktualizowanie | 67 |
| Instalacje nadzorowane | 67 |
| Instalacja nienadzorowana | 70 |
| Instalowanie usług Analysis Services | 76 |
| Tryb UDM | 78 |
| Tryb tabelaryczny | 78 |
| Instalowanie funkcji PowerPivot for SharePoint | 80 |
| Sprawdzanie systemu | 81 |
| Konfiguracja po instalacji | 81 |
| Ustawianie opcji systemu SQL Server pod kątem wydajności | 81 |
| Baza tempdb | 83 |
| Konfigurowanie ustawień systemu SQL Server w obszarze bezpieczeństwa | 85 |
| Narzędzie SQL Server Configuration Manager | 86 |
| Tworzenie kopii zapasowej | 86 |
| Usuwanie instalacji systemu SQL Server | 87 |
| Usuwanie instalacji usług Reporting Services | 87 |
| Usuwanie instalacji usług Analysis Services | 87 |
| Usuwanie instalacji silnika bazodanowego systemu SQL Server | 88 |
| Rozwiązywanie problemów z nieudaną instalacją | 88 |
| Podsumowanie | 88 |

Rozdział 3. Najlepsze praktyki aktualizowania systemu

| | |
|---|-----------|
| do wersji SQL Server 2014 | 91 |
| Po co aktualizować system do wersji SQL Server 2014? | 92 |
| Ograniczanie ryzyka — wkład Microsoftu | 93 |
| Niezależni producenci oprogramowania i wkład społeczności skupionej wokół systemu SQL Server | 93 |
| Aktualizowanie systemu do wersji SQL Server 2014 | 94 |
| Aktualizacja w miejscu | 94 |
| Aktualizowanie przy równoległe zainstalowanych wersjach | 96 |
| Wybór między aktualizowaniem w miejscu a podejściem z równoległymi egzemplarzami | 97 |
| Kroki i narzędzia potrzebne przed instalacją | 97 |
| Zadania wykonywane przed aktualizacją | 97 |
| Narzędzia przydatne przed aktualizacją | 98 |
| Zgodność wstecz | 106 |
| Funkcje nieobsługiwane lub wycofane w wersji SQL Server 2014 | 106 |
| Funkcje baz danych uznane za przestarzałe w wersji SQL Server 2014 | 108 |
| Inne zmiany w działaniu funkcji w wersji SQL Server 2014 | 108 |

| | |
|---|------------|
| Uwagi dotyczące komponentów systemu SQL Server | 109 |
| Aktualizowanie katalogu pełnotekstowego | 109 |
| Aktualizowanie usług Reporting Services | 109 |
| Aktualizowanie do wersji 64-bitowej | 110 |
| Testy po aktualizacji | 110 |
| Podsumowanie | 111 |
| Rozdział 4. Zarządzanie silnikiem bazodanowym i rozwiązywanie związanych z nim problemów | 113 |
| Narzędzia konfiguracyjne i administracyjne | 114 |
| SQL Server Configuration Manager | 114 |
| Parametry uruchomieniowe | 116 |
| Uruchomieniowe procedury składowane | 118 |
| Częściowo niezależne bazy danych | 120 |
| Narzędzia do rozwiązywania problemów | 121 |
| Połączenie DAC | 121 |
| Odtwarzanie systemowych baz danych | 122 |
| SQL Server Management Studio | 123 |
| Raporty | 124 |
| Konfigurowanie systemu SQL Server w narzędziu SQL Server Management Studio | 126 |
| Filtrowanie obiektów | 131 |
| Dzienniki błędów | 131 |
| Narzędzie Activity Monitor | 132 |
| Monitorowanie procesów w języku T-SQL | 136 |
| Procedury sp_who i sp_who2 | 137 |
| Widok sys.dm_exec_connections | 137 |
| Funkcja sys.dm_exec_sql_text | 138 |
| Zarządzanie wieloma serwerami | 138 |
| Centralne serwery zarządzania i grupy serwerów | 139 |
| Opcje śledzenia | 140 |
| Udzielanie pomocy przez obsługę techniczną | 141 |
| Narzędzie SQLDumper.exe | 141 |
| Narzędzie SQLDiag.exe | 142 |
| Podsumowanie | 144 |
| Rozdział 5. Automatyzowanie pracy systemu SQL Server | 147 |
| Plany konserwacji | 148 |
| Kreator planów konserwacji | 148 |
| Projektant planów konserwacji | 153 |
| Automatyzowanie systemu SQL Server za pomocą narzędzia SQL Server Agent | 156 |
| Zadania | 156 |
| Harmonogramy | 162 |
| Operatorzy | 163 |
| Alerty | 166 |

| | |
|--|------------|
| Zabezpieczenia w narzędziu SQL Server Agent | 171 |
| Konto usługowe | 171 |
| Dostęp do narzędzia SQL Server Agent | 172 |
| Jednostki pośredniczące narzędzia SQL Server Agent | 173 |
| Konfigurowanie narzędzia SQL Server Agent | 176 |
| Właściwości z kategorii General | 176 |
| Właściwości z kategorii Advanced | 177 |
| Właściwości z kategorii Alert System | 178 |
| Właściwości z kategorii Job System | 179 |
| Właściwości z kategorii Connection | 180 |
| Właściwości z kategorii History | 180 |
| Zarządzanie wieloma serwerami | 180 |
| Podstawianie tokenów | 181 |
| Przekazywanie zdarzeń | 183 |
| Używanie usług WMI | 184 |
| Zarządzanie wieloma serwerami | |
| — używanie serwerów nadrzędnych i docelowych | 185 |
| Podsumowanie | 187 |
| Rozdział 6. Service Broker w systemie SQL Server 2014 | 189 |
| Asynchroniczne przesyłanie komunikatów | 189 |
| Ogólne omówienie Service Brokera | 191 |
| Service Broker a inne kolejki komunikatów | 192 |
| Konfigurowanie Service Brokera | 193 |
| Ustawianie stanu Service Brokera | 193 |
| Typy komunikatów | 195 |
| Kontrakty | 196 |
| Kolejki | 197 |
| Usługi | 199 |
| Trasy | 199 |
| Priorytety | 201 |
| Grupy konwersacji | 202 |
| Używanie Service Brokera | 203 |
| Wysyłanie komunikatów | 203 |
| Odbieranie komunikatów | 206 |
| Przesyłanie komunikatów między egzemplarzami | 207 |
| Zewnętrzna aktywacja | 209 |
| Przykład ilustrujący zapisywanie danych o użytkownikach w dzienniku | 210 |
| Podsumowanie | 217 |
| Rozdział 7. Integrowanie systemu SQL Server ze środowiskiem CLR | 219 |
| Wprowadzenie do środowiska CLR | 220 |
| System SQL Server jako host środowiska | |
| uruchomieniowego platformy .NET | 221 |
| Domeny aplikacji | 221 |
| Język T-SQL a środowisko CLR | 222 |
| Włączanie integracji ze środowiskiem CLR | 222 |

| | |
|--|------------|
| Tworzenie komponentów CLR | 223 |
| Metoda bez używania środowiska Visual Studio | 224 |
| Używanie narzędzi Microsoft SQL Server Data Tools | 226 |
| Zabezpieczenia przy integracji ze środowiskiem CLR | 227 |
| Monitorowanie wydajności | 228 |
| Windows System Monitor | 228 |
| Rozszerzone zdarzenia | 230 |
| Widoki DMV | 231 |
| Cele projektowe dotyczące integracji ze środowiskiem CLR | 231 |
| Podsumowanie | 232 |
| Rozdział 8. Zabezpieczanie egzemplarzy baz danych | 233 |
| Rodzaje uwierzytelniania | 234 |
| Uwierzytelnianie z systemu SQL Server | 234 |
| Uwierzytelnianie z systemu Windows | 235 |
| Porównanie obu metod uwierzytelniania | 236 |
| Loginy i użytkownicy | 236 |
| Autoryzacja dostępu | 237 |
| Uprawnienia na serwerze | 237 |
| Zabezpieczane obiekty baz danych | 243 |
| Łańcuchy uprawnień | 244 |
| Łańcuchy uprawnień obowiązujące między bazami | 245 |
| Zabezpieczenia na poziomie wierszy | 247 |
| Podsumowanie | 248 |
| Rozdział 9. Technologia In-Memory OLTP | 249 |
| Stosowanie i implementowanie technologii In-Memory OLTP | 249 |
| Włączanie technologii In-Memory OLTP | 250 |
| Struktury tabel używanych w technologii In-Memory OLTP | 252 |
| Rekordy | 252 |
| Indeksy | 253 |
| Kwestie związane z procesorem | 258 |
| Zagadnienia związane z wirtualizacją | 258 |
| Kwestie związane z pamięcią | 258 |
| Zarządzanie pamięcią za pomocą funkcji Resource Governor | 261 |
| Tworzenie natywnie kompilowanych procedur składowanych | 263 |
| Omówienie narzędzia ARM | 266 |
| Podsumowanie | 278 |
| Rozdział 10. Konfigurowanie serwera pod kątem optymalnej wydajności | 279 |
| Co każdy administrator baz danych powinien | |
| wiedzieć na temat wydajności? | 280 |
| Cykl dostrajania wydajności | 280 |
| Konfiguracja | 282 |
| Plan zasilania | 282 |
| Natychmiastowe inicjowanie plików bazy danych | 284 |

| | |
|---|------------|
| Opcje śledzenia | 287 |
| Definicja dobrej wydajności | 287 |
| Koncentracja na tym, co najważniejsze | 288 |
| Co administrator programista powinien wiedzieć na temat wydajności? | 289 |
| Użytkownicy | 289 |
| Instrukcje w języku SQL | 290 |
| Wzorce używania danych | 290 |
| Schemat bazy danych | 290 |
| Co o wydajności powinien wiedzieć administrator | |
| produkcyjnej bazy danych? | 291 |
| Optymalizowanie sprzętu serwera | 292 |
| Zarządzanie sprzętem | 293 |
| Procesor | 294 |
| Architektura x64 | 294 |
| Pamięć podręczna | 295 |
| Technologia Hyper-Threading | 295 |
| Pojęcia związane z systemami wielordzeniowymi | 296 |
| Pamięć | 297 |
| Pamięć fizyczna | 297 |
| Fizyczna przestrzeń adresowa | 298 |
| Menedżer pamięci wirtualnej | 298 |
| Plik stronicowania | 299 |
| Błędy strony | 300 |
| Operacje wejścia-wyjścia | 300 |
| Sieć | 301 |
| Dyski magnetyczne | 302 |
| Dyski SSD (dyski z pamięcią flash) | 305 |
| Rozważania związane z przechowywaniem danych | 305 |
| Podsumowanie | 309 |
| Rozdział 11. Optymalizowanie systemu SQL Server 2014 | 311 |
| Optymalizowanie aplikacji | 311 |
| Definiowanie obciążenia | 312 |
| Cichy zabójca — problemy z operacjami wejścia-wyjścia | 312 |
| Model obsługi wejścia-wyjścia w systemie SQL Server | 312 |
| Rozmieszczanie plików bazy danych | 313 |
| Kwestie związane z bazą tempdb | 314 |
| Wewnętrzne mechanizmy systemu SQL Server i alokacja plików | 318 |
| Podział tabel i indeksów na partycje | 320 |
| Po co stosować partycje? | 320 |
| Tworzenie funkcji partycjonującej | 322 |
| Tworzenie grup plików | 324 |
| Tworzenie schematu partycjonowania | 324 |
| Kompresja danych | 325 |
| Kompresja wierszy | 326 |
| Kompresja stron | 328 |

| | |
|---|-----|
| Szacowanie oszczędności miejsca | 329 |
| Monitorowanie kompresji danych | 331 |
| Uwagi dotyczące kompresji danych | 332 |
| System SQL Server i procesory | 333 |
| Architektura NUMA i dynamicznie dodawane procesory | 334 |
| Spójność pamięci podręcznej | 334 |
| Maska koligacji | 336 |
| Ustawienie Max Degree of Parallelism (MAXDOP) | 336 |
| Ustawienie Cost Threshold for Parallelism | 337 |
| Uwagi na temat pamięci i związanych z nią usprawnień | 339 |
| Rozszerzenia puli buforów | 339 |
| Dostrajanie pamięci systemu SQL Server | 341 |
| Lokalność danych | 343 |
| Maksymalna ilość pamięci dla serwera | 343 |
| Resource Governor | 344 |
| Podstawowe elementy technologii Resource Governor | 345 |
| Używanie technologii Resource Governor w programie SQL Server 2014 Management Studio | 349 |
| Monitorowanie pracy mechanizmu Resource Governor | 350 |
| Podsumowanie | 351 |

Rozdział 12. Monitorowanie systemu SQL Server353

| | |
|---|-----|
| Cele monitorowania | 354 |
| Określanie celów monitorowania | 354 |
| Określanie punktu odniesienia | 355 |
| Porównywanie aktualnych wartości z punktem odniesienia | 355 |
| Wybór odpowiednich narzędzi monitorujących | 356 |
| Monitor wydajności | 358 |
| Liczniki dotyczące zasobów procesora | 359 |
| Aktywność dysków | 360 |
| Użytkowanie pamięci | 366 |
| Narzędzia do monitorowania wydajności | 369 |
| Monitorowanie zdarzeń | 371 |
| Domyślny ślad | 373 |
| Sesja system_health | 374 |
| SQL Trace | 374 |
| Powiadomienia o zdarzeniach | 378 |
| Zdarzenia rozszerzone systemu SQL Server | 381 |
| Monitorowanie systemu za pomocą widoków DMV i funkcji DMF | 400 |
| Co dzieje się w systemie SQL Server? | 401 |
| Wyświetlanie informacji o blokadach z bazy danych | 404 |
| Wyświetlanie informacji o blokadach z serwera | 405 |
| Używanie indeksu w bazie danych | 405 |
| Indeksy nieużywane w bazie danych | 406 |
| Wyświetlanie kwerend oczekujących na pamięć | 407 |

| | |
|---|-----|
| Informacje o podłączonych użytkownikach | 408 |
| Wolna przestrzeń w grupach plików | 408 |
| Plan wykonywania i kod aktualnie działających kwerend | 409 |
| Zajmowana pamięć | 409 |
| Zajęta pamięć w puli buforów | 409 |
| Monitorowanie dzienników | 410 |
| Monitorowanie dziennika błędów systemu SQL Server | 410 |
| Monitorowanie dzienników zdarzeń systemu Windows | 411 |
| Standardowe raporty systemu SQL Server | 411 |
| Narzędzie System Center Advisor | 411 |
| Podsumowanie | 414 |

Rozdział 13. Dostrajanie wydajności kodu w języku T-SQL 415

| | |
|---|-----|
| Omówienie procesu przetwarzania kwerend | 416 |
| Generowanie planów wykonywania | 418 |
| Statystyki | 418 |
| Usprawnienia optymalizatora kwerend w systemie SQL Server 2014 | |
| — nowy mechanizm szacowania kardynalności | 419 |
| Używanie nowego mechanizmu szacowania kardynalności | 419 |
| Identyfikowanie problemów przy dostrajaniu | |
| wydajności kwerend w języku SQL | 420 |
| Monitorowanie wydajności kwerend | 420 |
| Co robić po wykryciu kwerendy o niskiej wydajności? | 421 |
| Generowanie planów wykonywania kwerend | 423 |
| Czytanie planów wykonywania kwerend | 424 |
| Operatory dostępu do danych w planach wykonywania kwerend | 427 |
| Skanowanie tabeli | 428 |
| Skanowanie indeksów klastrowanych | 429 |
| Skanowanie indeksów nieklastrowanych | 432 |
| Łączenie operatorów dostępu | 434 |
| Operatory złączeń | 435 |
| Pętla zagnieżdżona (złączanie w pętli) | 435 |
| Złączenia z haszowaniem | 437 |
| Złączanie przez scalanie | 438 |
| Plany wykonywania kwerend modyfikujących dane | 440 |
| Przetwarzanie kwerend działających na tabelach i indeksach | |
| podzielonych na partycje | 442 |
| Operacje, na które wpływa podział na partycje | 442 |
| Strategie współbieżnego wykonywania kwerend | |
| dla obiektów podzielonych na partycje | 444 |
| Analizowanie wydajności kwerend w środowisku produkcyjnym | 444 |
| Systemowe widoki DMV | 445 |
| Łączenie wszystkich elementów | 446 |
| Podsumowanie | 447 |

| | |
|---|------------|
| Rozdział 14. Indeksowanie baz danych | 449 |
| Nowe mechanizmy w indeksach w systemie SQL Server 2014 | 450 |
| Indeksy i tabele podzielone na partycje | 454 |
| Omówienie indeksów | 454 |
| Tworzenie indeksów | 459 |
| Używanie tabel i indeksów podzielonych na partycje | 461 |
| Konserwacja indeksów | 461 |
| Monitorowanie poziomu fragmentacji indeksu | 462 |
| Porządkowanie indeksów | 463 |
| Zwiększanie wydajności kwerend za pomocą indeksów | 464 |
| Narzędzie DTA | 469 |
| Podsumowanie | 472 |
| Rozdział 15. Replikacja | 473 |
| Wprowadzenie do procesu replikacji | 474 |
| Elementy w procesie replikacji | 474 |
| Typy replikacji | 476 |
| Modele replikacji | 478 |
| Jeden wydawca i dowolna liczba subskrybentów | 478 |
| Wielu wydawców i jeden subskrybent | 480 |
| Wielu wydawców będących jednocześnie subskrybentami | 480 |
| Subskrybent aktualizujący dane | 481 |
| Model P2P | 481 |
| Implementowanie replikacji | 483 |
| Konfigurowanie replikacji migawkowej | 483 |
| Konfigurowanie dystrybucji | 483 |
| Implementowanie replikacji migawkowej | 487 |
| Implementowanie replikacji transakcyjnej i ze skalaniem | 500 |
| Replikacja w trybie P2P | 501 |
| Konfigurowanie replikacji w trybie P2P | 501 |
| Konfigurowanie architektury dla replikacji P2P | 502 |
| Replikacja obsługiwana za pomocą skryptów | 505 |
| Monitorowanie replikacji | 506 |
| Monitor replikacji | 506 |
| Monitor wydajności | 508 |
| Widoki DMV związane z replikacją | 508 |
| Procedura sp_replcounters | 509 |
| Podsumowanie | 509 |
| Rozdział 16. Klastrowanie w systemie SQL Server 2014 | 511 |
| Klastrowanie a sytuacja firmy | 512 |
| Co można uzyskać przy użyciu klastrów? | 512 |
| Czego klastrowanie nie umożliwia? | 513 |
| Właściwe powody stosowania klastrowania | |
| dla systemu SQL Server 2014 | 514 |
| Co zastosować zamiast klastrowania? | 514 |

| | |
|--|-----|
| Klastrowanie — ogólne omówienie | 517 |
| Jak działa klastrowanie? | 517 |
| Rodzaje klastrów | 522 |
| Aktualizowanie klastrów z systemem SQL Server | 524 |
| Rezygnacja z aktualizacji | 524 |
| Aktualizowanie klastra do wersji SQL Server 2014 w miejscu | 524 |
| Tworzenie klastra od nowa | 525 |
| Plan wycofywania zmian | 527 |
| Która metoda aktualizacji jest najlepsza? | 527 |
| Przygotowanie do klastrowania | 527 |
| Przygotowywanie infrastruktury | 528 |
| Przygotowywanie sprzętu | 528 |
| Tworzenie klastra w systemie operacyjnym Windows Server 2012 R2 | 531 |
| Co trzeba zrobić przed instalacją klastra w systemie Windows 2012 R2? | 531 |
| Instalowanie klastra WSFC z systemu Windows Server 2012 R2 | 532 |
| Przygotowywanie systemu Windows Server 2012 R2 do pracy w klastrze | 535 |
| Używanie koordynatora MSDTC w klastrze | 536 |
| Dodawanie systemu SQL Server 2014 do klastra | 537 |
| Instrukcje instalowania systemu SQL Server w klastrze | 538 |
| Instalowanie pakietów poprawek SP | |
| i aktualizacji typu Cumulative Update | 544 |
| Testuj, testuj i jeszcze raz testuj | 544 |
| Zarządzanie klastrem i monitorowanie go | 546 |
| Rozwiązywanie problemów z klastrem | 547 |
| Jak podejść do rozwiązywania problemów z klastrami WSFC? | 547 |
| Właściwe rozwiązanie już za pierwszym razem | 548 |
| Zbieranie informacji | 548 |
| Rozwiązywanie problemów | 549 |
| Współpraca z Microsoftem | 549 |
| Podsumowanie | 549 |

Rozdział 17. Kopie zapasowe i przywracanie stanu551

| | |
|---|-----|
| Usprawnienia związane z tworzeniem kopii zapasowej | |
| i przywracaniem stanu | 552 |
| Tworzenie kopii zapasowych z zapisem na adres URL | 552 |
| Zarządzane tworzenie kopii zapasowych | |
| z zapisem w platformie Windows Azure | 552 |
| Szyfrowanie kopii zapasowych | 553 |
| Przegląd procesu tworzenia kopii zapasowej i przywracania stanu | 553 |
| Jak przebiega tworzenie kopii zapasowej? | 553 |
| Kopiowanie baz danych | 556 |
| Kompresja kopii zapasowej | 568 |
| Modele odzyskiwania | 569 |
| Wybór modelu odzyskiwania | 571 |
| Przełączanie modeli odzyskiwania | 572 |

| | |
|--|------------|
| Tworzenie tabel z historią kopii zapasowych | 573 |
| Uprawnienia potrzebne do tworzenia kopii zapasowych i przywracania stanu | 574 |
| Tworzenie kopii zapasowych systemowych baz danych | 575 |
| Kopie zapasowe indeksów pełnotekstowych | 576 |
| Sprawdzanie poprawności obrazów z kopią zapasową | 576 |
| Przebieg przywracania | 577 |
| Przygotowania do odzyskiwania | 579 |
| Wymagania związane z odzyskiwaniem | 579 |
| Wzorce użytkownika danych | 580 |
| Czas na konserwację | 581 |
| Inne rozwiązania zapewniające wysoką dostępność | 582 |
| Opracowywanie i wykonywanie planu tworzenia kopii zapasowych | 583 |
| Używanie programu SQL Server Management Studio | 583 |
| Plany konserwacji bazy danych | 587 |
| Używanie instrukcji tworzenia kopii zapasowych z języka T-SQL | 591 |
| Zarządzanie kopiami zapasowymi | 593 |
| Wydajność tworzenia i przywracania kopii zapasowych | 593 |
| Odzyskiwanie baz danych | 594 |
| Proces przywracania | 594 |
| Przywracanie baz za pomocą programu SQL Server Management Studio | 598 |
| Polecenie RESTORE z języka T-SQL | 602 |
| Przywracanie systemowych baz danych | 603 |
| Archiwizowanie danych | 604 |
| Podział tabel w systemie SQL Server na partycje | 605 |
| Widoki podzielone na partycje | 606 |
| Podsumowanie | 606 |
| Rozdział 18. Przesyłanie dzienników w systemie SQL Server 2014 | 607 |
| Scenariusze stosowania przesyłania dzienników | 608 |
| Przesyłanie dzienników w celu utworzenia aktywnego serwera rezerwowego | 608 |
| Przesyłanie dzienników jako mechanizm odzyskiwania systemu po katastrofie | 609 |
| Przesyłanie dzienników jako mechanizm tworzenia baz potrzebnych do generowania raportów | 610 |
| Architektura mechanizmu przesyłania dzienników | 611 |
| Serwer główny | 612 |
| Serwer pomocniczy | 612 |
| Serwer monitorujący | 612 |
| Proces przesyłania dzienników | 613 |
| Wymagania systemowe | 613 |
| Sieć | 613 |
| Serwery o identycznych możliwościach | 614 |
| Przechowywanie danych | 614 |
| Oprogramowanie | 614 |

| | |
|--|------------|
| Wdrażanie przesyłania dzienników | 615 |
| Wstępne konfigurowanie | 615 |
| Wdrażanie przesyłania dzienników za pomocą programu SQL Server Management Studio | 616 |
| Wdrażanie przesyłania dzienników za pomocą instrukcji języka T-SQL | 624 |
| Monitorowanie systemu i rozwiązywanie problemów | 625 |
| Monitorowanie z wykorzystaniem programu SQL Server Management Studio | 626 |
| Monitorowanie z wykorzystaniem procedur składowanych | 626 |
| Rozwiązywanie problemów | 627 |
| Zarządzanie zmianą ról | 627 |
| Synchronizowanie zależności | 627 |
| Przełączanie ról między serwerami głównym i pomocniczym | 630 |
| Przełączanie między serwerami głównym i pomocniczym | 632 |
| Przekierowywanie klientów na serwer pomocniczy | 633 |
| Plan tworzenia kopii zapasowych baz danych | 634 |
| Integrowanie przesyłania dzienników z innymi rozwiązaniami zapewniającymi wysoką dostępność | 635 |
| Kopie lustrzane danych w systemie SQL Server 2014 | 635 |
| Klastry WSFC | 635 |
| Replikacja w systemie SQL Server 2014 | 636 |
| Kończenie przesyłania dzienników | 637 |
| Kończenie przesyłania dzienników w programie SQL Server Management Studio | 637 |
| Kończenie przesyłania dzienników za pomocą języka T-SQL | 637 |
| Wydajność przesyłania dzienników | 638 |
| Aktualizacja przesyłania dzienników do wersji z systemu SQL Server 2014 | 639 |
| Podejście z minimalnym przestojem | 639 |
| Podejście ze standardowym przestojem | 639 |
| Podejście z instalowaniem przesyłania dzienników | 640 |
| Podsumowanie | 640 |
| Rozdział 19. Kopie lustrzane baz danych | 643 |
| Wprowadzenie do kopii lustrzanych baz danych | 644 |
| Tryby działania kopii lustrzanych | 645 |
| Przykłady tworzenia kopii lustrzanych baz danych | 648 |
| Przygotowywanie punktów końcowych | 648 |
| Przygotowywanie bazy danych do tworzenia kopii lustrzanych | 653 |
| Początkowa synchronizacja serwerów głównego i lustrzanego | 654 |
| Uruchamianie sesji tworzenia kopii lustrzanych | 655 |
| Tryb wysokiego bezpieczeństwa bez automatycznego przełączania awaryjnego | 656 |
| Tryb wysokiego bezpieczeństwa z automatycznym przełączaniem awaryjnym | 657 |
| Tryb wysokiej wydajności | 658 |

| | |
|---|-----|
| Kopie lustrzane i różne edycje systemu SQL Server 2014 | 659 |
| Zmiana ról w mechanizmie tworzenia kopii lustrzanych | 659 |
| Automatyczne przełączanie awaryjne | 660 |
| Ręczne przełączanie awaryjne | 662 |
| Wymuszone przełączenie awaryjne | 664 |
| Używanie monitora tworzenia kopii lustrzanych | 665 |
| Ustawianie wartości progowych liczników i wysyłanie alertów | 668 |
| Przygotowywanie serwera lustrzanego do przełączenia awaryjnego | 670 |
| Sprzęt, oprogramowanie i konfiguracja serwera | 670 |
| Dostępność bazy danych w trakcie planowanego przestoju | 672 |
| Konfigurowanie zadań narzędzia SQL Server Agent na serwerze lustrzanym | 673 |
| Przekierowywanie klientów na serwer lustrzany | 674 |
| Tworzenie kopii lustrzanych baz i inne metody zapewniania wysokiej dostępności | 675 |
| Kopie lustrzane baz danych a klastry | 675 |
| Kopie lustrzane baz danych a replikacja transakcyjna | 676 |
| Kopie lustrzane baz danych a przesyłanie dzienników | 676 |
| Kopie lustrzane baz danych a grupy dostępności | 676 |
| Snapshoty baz danych | 677 |
| Podsumowanie | 678 |

Rozdział 20. Zarządzanie usługami Integration Services i dostrajanie ich wydajności 679

| | |
|--|-----|
| Przegląd usług SSIS | 680 |
| Zastosowania usług SSIS | 680 |
| Główne elementy usług SSIS | 681 |
| Zarządzanie projektem i kontrola zmian | 683 |
| Zarządzanie usługą SSIS | 683 |
| Wprowadzenie do usługi SSIS | 683 |
| Konfiguracja | 684 |
| Dzienniki zdarzeń | 687 |
| Monitorowanie aktywności | 688 |
| Zarządzanie pakietami SSIS w modelu z instalowaniem pakietów | 688 |
| Zarządzanie pakietami za pomocą programu SQL Server Management Studio | 689 |
| Instalowanie | 691 |
| Zarządzanie pakietami SSIS w modelu z instalacją projektów | 694 |
| Konfigurowanie katalogu SSIS | 695 |
| Instalowanie pakietów | 696 |
| Konfigurowanie pakietów | 697 |
| Wykonywanie i planowanie uruchamiania pakietów | 701 |
| Uruchamianie pakietów w narzędziu SSDT | 701 |
| Uruchamianie pakietów za pomocą kreatora Import and Export Wizard | 701 |
| Uruchamianie pakietów za pomocą narzędzia DTEXEC | 702 |

| | |
|---|-----|
| Uruchamianie pakietów za pomocą narzędzia DTExecUI (w modelu z instalowaniem pakietów) | 702 |
| Uruchamianie pakietów za pomocą narzędzia Execute Package (w modelu z instalowaniem projektów) | 703 |
| Planowanie wykonywania za pomocą narzędzia SQL Server Agent | 704 |
| Uruchamianie pakietów za pomocą języka T-SQL | 706 |
| Zabezpieczanie usług SSIS | 706 |
| Przegląd zabezpieczeń usług SSIS | 706 |
| Zabezpieczenia w modelu z instalowaniem pakietów | 707 |
| Podsumowanie | 709 |

Rozdział 21. Zarządzanie usługami Analysis Services i dostrajanie ich wydajności 711

| | |
|--|-----|
| Przegląd usług SSAS | 712 |
| Elementy modelu MOLAP | 713 |
| Elementy modelu tabelowego | 713 |
| Elementy architektury serwera SSAS | 714 |
| Zarządzanie serwerem SSAS | 715 |
| Właściwości serwera | 715 |
| Wymagane usługi | 717 |
| Język ASSL | 717 |
| Zarządzanie bazami z serwera SSAS | 719 |
| Instalowanie baz na serwerach SSAS | 719 |
| Przetwarzanie obiektów SSAS | 721 |
| Tworzenie i przywracanie kopii zapasowych baz SSAS | 727 |
| Synchronizowanie baz SSAS | 729 |
| Monitorowanie i dostrajanie wydajności usług SSAS | 730 |
| Monitorowanie zdarzeń SSAS | 731 |
| Używanie narzędzia Flight Recorder do analiz następczych | 731 |
| Podsumowanie | 732 |

Rozdział 22. Zarządzanie usługami Reporting Services w systemie SQL Server 733

| | |
|--|-----|
| Narzędzie Configuration Manager usług SSRS | 734 |
| Konto usługowe | 736 |
| Adres URL usługi sieciowej | 737 |
| Bazy SSRS | 739 |
| Adres URL narzędzia Report Manager | 741 |
| Ustawienia poczty elektronicznej | 742 |
| Konto wykonawcze | 743 |
| Klucze szyfrowania | 744 |
| Skalowanie systemu | 745 |
| Dziennik wykonywania raportów | 746 |
| Report Manager | 747 |
| Używanie narzędzia Report Manager | 747 |
| Zarządzanie raportami | 754 |
| Podsumowanie | 767 |

| | |
|--|------------|
| Rozdział 23. Integracja systemu SQL Server 2014 z platformą SharePoint 2013 | 769 |
| Komponenty uczestniczące w integracji | 770 |
| PowerPivot | 770 |
| Usługi Reporting Services | 772 |
| Power View | 775 |
| Odświeżanie danych | 776 |
| Używanie połączeń z danymi w Excelu | 777 |
| Połączenia danych z narzędzia PerformancePoint | 782 |
| Odświeżanie danych w usługach Visio | 783 |
| Odświeżanie danych w dodatku PowerPivot | 785 |
| Podsumowanie | 794 |
| Rozdział 24. Administrowanie bazami SQL Database i ich konfigurowanie | 795 |
| Wprowadzenie do technologii Windows Azure SQL Database | 795 |
| Architektura bazy SQL Database | 796 |
| Warstwa klientów | 797 |
| Warstwa usług | 797 |
| Warstwa platformy | 797 |
| Warstwa infrastruktury | 797 |
| Różnice między środowiskami | 797 |
| Konfigurowanie baz SQL Database | 798 |
| Udostępnianie serwera i bazy danych | 798 |
| Ograniczanie dostępu do zasobów i równoważenie obciążenia | 805 |
| Konfigurowanie zapory dla baz SQL Database | 806 |
| Nawiązywanie połączenia z bazami SQL Database | 807 |
| Administrowanie bazami SQL Database | 809 |
| Tworzenie loginów i kont użytkowników | 809 |
| Przypisywanie uprawnień dostępu | 811 |
| Praca z bazami SQL Database | 812 |
| Tworzenie kopii zapasowych baz SQL Database | 813 |
| Okno Object Explorer dla baz SQL Database | 813 |
| Czego brakuje w bazach SQL Database? | 815 |
| Podsumowanie | 816 |
| Rozdział 25. Grupy dostępności AlwaysOn | 817 |
| Architektura | 818 |
| Repliki i role w grupach dostępności | 819 |
| Tryby dostępności | 819 |
| Obsługiwane typy przełączania awaryjnego | 820 |
| Umożliwianie dostępu do replik pomocniczych | |
| w trybie tylko do odczytu | 821 |
| Przykład ilustrujący grupę dostępności | 822 |
| Konfigurowanie nowej grupy dostępności | 823 |
| Konfigurowanie istniejących grup dostępności | 830 |
| Przełączanie awaryjne w grupach dostępności | 834 |
| Zawieszanie pracy baz wysokiej dostępności | 836 |

| | |
|---|------------|
| Wznawianie pracy baz wysokiej dostępności | 837 |
| Połączenia aplikacji klienckich | 837 |
| Aktywna replika pomocnicza udostępniana w trybie tylko do odczytu | 839 |
| Dostęp w trybie tylko do odczytu | 839 |
| Łączenie się klientów z repliką pomocniczą | 840 |
| Wydajność | 841 |
| Tworzenie kopii zapasowych w replice pomocniczej | 842 |
| Analizowanie metadanych z replik, w których wykonywane są kopie zapasowe | 843 |
| Panel kontrolny grup dostępności AlwaysOn | 844 |
| Monitorowanie i rozwiązywanie problemów | 846 |
| Podsumowanie | 847 |
| Skorowidz..... | 849 |

Konfigurowanie serwera pod kątem optymalnej wydajności

ZAWARTOŚĆ ROZDZIAŁU:

- Definiowanie wysokiej wydajności.
- Co każdy administrator baz danych powinien wiedzieć o wydajności?
- Konfigurowanie sprzętu serwera.
- Szczegółowe omówienie konfiguracji procesora.
- Omówienie konfiguracji pamięci i opcji z tego obszaru.
- Projekt i opcje dotyczące operacji wejścia-wyjścia.

W branży informatycznej za bazy danych i systemy, w których te bazy się znajdują, odpowiadają osoby pełniące wiele różnych ról. *Administrator programista* odpowiada głównie za projektowanie bazy, a także za pisanie kodu (kwerend, procedur składowanych itd.). *Administrator produkcyjnej bazy danych* zajmuje się przede wszystkim konfiguracją, konserwacją i zapewnieniem dostępności bazy oraz systemu. *Administrator procesów BI* odpowiada głównie za procesy BI oparte na systemie SQL Server i powiązanych narzędziach. Jedna osoba może nawet odpowiadać za kilka tych zadań. Wtedy można ją nazwać *hybrydowym administratorem baz danych*. Ponadto niektórzy administratorzy baz danych nie posiadają formalnego wykształcenia z tego zakresu lub zajmują się serwerami z konieczności, ponieważ brakuje odpowiednich pracowników. Ich można określić mianem *okazyjnych administratorów baz danych*. Osoby te zwykle pełnią wiele obowiązków i z powodu braku zasobów muszą szybko wdrożyć się do pracy. Szczegółowe omówienie różnych profesjonalistów zajmujących się bazami danych znajdziesz w rozdziale 1., „Architektura systemu SQL Server 2014”.

Administrator programista musi wiedzieć, jak optymalizować wydajność systemu, aby projektowane rozwiązania wykorzystywały potencjał sprzętu. Osoby takie muszą zadbać o wydajną pracę systemów także po wprowadzeniu w nich zmian (które są nieuniknione). Proces rozrastania się systemów (ze względu na ilość danych, liczbę użytkowników i nowe funkcje) powinien przebiegać w taki sposób, by cały czas działały one optymalnie.

Także administrator produkcyjnych baz danych musi rozumieć kwestie związane z wydajnością, aby zarządzany przez niego system działał poprawnie zarówno na początku, jak i w całym cyklu życia. Należy przy tym uwzględnić kilka czynników, takich jak poprawna konfiguracja serwera, śledzenie pracy systemu w momencie uruchomienia, wdrożenie kompletnej strategii monitorowania pozwalającej zapewnić optymalne działanie systemu itd.

Poniżej wymieniono najważniejsze aspekty związane z zapewnianiem wysokiej wydajności w zakresie skalowalności, czasu reakcji, niezawodności i używalności. Oto one.

- Zmiana konfiguracji zasilania ze zrównoważonej na wysoką wydajność.
- Włączenie funkcji Instant File Initialization w celu zminimalizowania liczby operacji wejścia-wyjścia na sekundę w trakcie powiększania plików.
- Zapoznanie się z opcjami śledzenia udostępnianymi przez Microsoft (opcje umożliwiają stosowanie domyślnie wyłączonych funkcji systemu SQL Server).
- Ustalenie możliwości systemu w zakresie szybkości procesora, pamięci i operacji wejścia-wyjścia.
- Wykrycie wąskich gardeł i ustalenie, jak rozwiązać te problemy.

W tym rozdziale omawiamy wszystkie wymienione kwestie oraz odpowiadamy na najważniejsze pytania dotyczące wydajności i konfigurowania serwera.

Co każdy administrator baz danych powinien wiedzieć na temat wydajności?

W tym rozdziale znajdziesz wiele rekomendacji, które umożliwią poprawienie wydajności systemu. Jednak przyspieszenie transakcji o kilka milisekund nie zawsze jest warte Twojego czasu i pieniędzy wydawanych na sprzęt. Często dobry początkowy plan jest wart więcej niż późniejsze pomysłowe optymalizacje. W kontekście wydajności zawsze pamiętaj o trzech rzeczach:

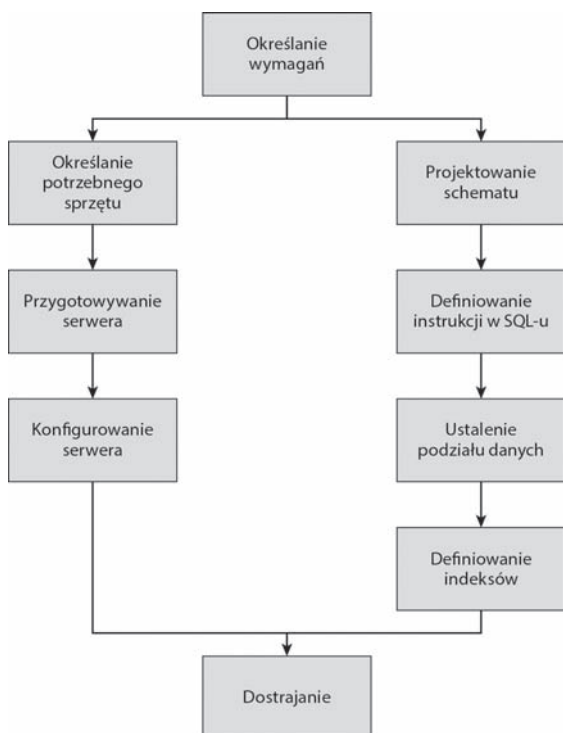
- cyklu dostrajania wydajności,
- definiowaniu dobrej wydajności,
- koncentrowaniu się na tym, co najważniejsze.

Cykl dostrajania wydajności

Zbyt często prace nad wydajnością i optymalizacją zostawia się na sam koniec. *Dostrajanie wydajności* to proces iteracyjny i najlepiej go zacząć już na początku projektowania systemu. Zapewnianie wysokiej wydajności rozpoczyna się od właściwego skonfigurowania serwera, po czym należy zaprojektować wydajny schemat, opracować szybkie instrukcje w SQL-u i odpowiednio dobrać indeksy. Monitorowanie i analiza wydajności zapewniają informacje, dzięki którym można wprowadzić zmiany w konfiguracji serwera, projekcie schematu i innych elementach.

Istnieje wiele narzędzi niezależnych producentów, które wspomagają monitorowanie egzemplarzy systemu SQL Server i środowiska. W samym systemie Microsoft SQL Server dostępne są bazy MDW (więcej informacji na ich temat podajemy w rozdziale 12., „Monitorowanie systemu SQL Server”). Korzystanie z narzędzi do monitorowania skryptów to ważne elementy procesu dostrajania wydajności.

Na rysunku 10.1 przedstawiono plan prac projektowych.



Rysunek 10.1. Prace projektowe

Przy tworzeniu nowych aplikacji nie istnieje system, w którym można przeprowadzić pomiary. W najlepszej sytuacji dostępne są wskaźniki oparte na obecnej grupie użytkowników lub prognozy zarządu określające, kim będą użytkownicy, co będą robili i jak wpłynie to na rozwijaną aplikację.

Jeśli system już istnieje i chcesz przenieść go na nowy serwer lub rozbudować, możesz wykonać pomiary wykorzystania zasobów systemowych i użyć wyników jako punktu wyjścia. Następnie możesz uwzględnić informacje na temat nowych funkcji. Odpowiedz przy tym na następujące pytania.

- Czy nastąpi wzrost liczby użytkowników?
- Czy zwiększy się obciążenie serwera?
- Czy zmieni się ilość danych?

Informacje te pomogą oszacować wpływ wprowadzenia nowego systemu na zasoby. Przed zaimplementowaniem nowego systemu, w trakcie testów, możesz zacząć porównywać szacunki z rzeczywistym zapotrzebowaniem na zasoby i wydajnością serwerów testowych.

Konfiguracja

Domyślna konfiguracja systemu SQL Server nie jest optymalna. Przed zainstalowaniem systemu i po zakończeniu tego procesu powinieneś wprowadzić zmiany w ustawieniach zasilania i zasadach grup, skonfigurować bazę tempdb, a także włączyć potrzebne opcje śledzenia. Kroki te wymagają zastosowania zaawansowanych opcji konfiguracyjnych wykraczających poza zakres rozdziału 2., „Najlepsze praktyki związane z instalowaniem systemu SQL Server 2014”.

Plan zasilania

Od wersji Windows 2003 w systemie Windows dostępne są plany zasilania. *Plan zasilania* określa, jak system operacyjny ma korzystać z energii. Każdy system operacyjny Windows ma trzy domyślne plany zasilania. Ponadto można tworzyć i instalować niestandardowe plany zasilania. Oto trzy domyślne plany.

Zrównoważony. W tym planie zasilanie jest dostosowane do potrzeb. Moc obliczeniowa procesora jest wykorzystywana w stopniu zależnym od ogólnego obciążenia powodowanego przez aplikację. Plan ten ma zapewniać wydajne zużycie pamięci przy minimalnym wpływie na wydajność.

Wysoka wydajność. To ustawienie zwiększa wydajność procesora i innego sprzętu serwera kosztem wyższego zużycia energii.

Oszczędzanie energii. Ten plan powoduje ograniczenie wydajności w celu zmniejszenia zużycia energii i kosztów pracy serwera. W odróżnieniu od poprzednich plan ten ogranicza szybkość procesora do określonego procentu jego możliwości.

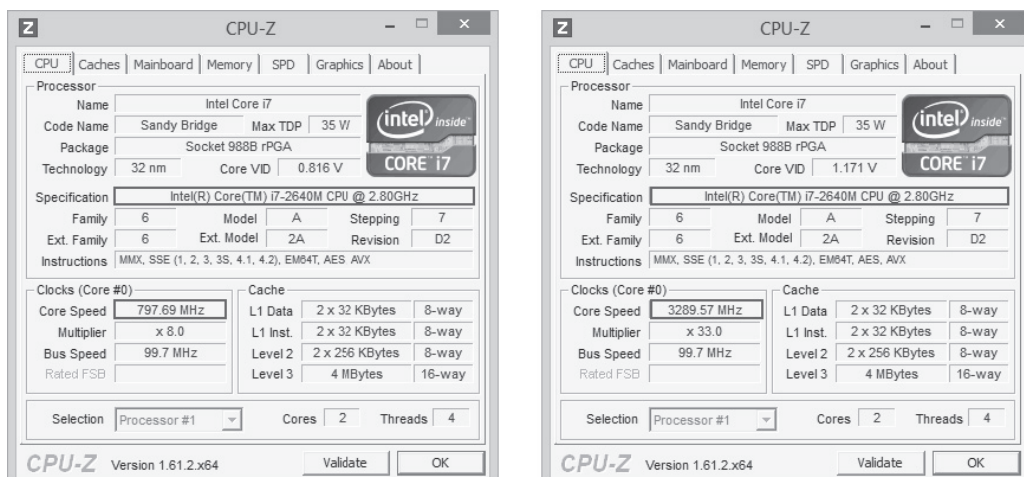
Od systemu Windows 2008 (i Windows Server 2012 R2) domyślnym planem zasilania jest Zrównoważony. Nie jest to korzystne dla systemu SQL Server. W opisie tego ustawienia znalazła się wzmianka o *minimalnym wpływie na wydajność*. Wszystko, co wpływa na wydajność systemu SQL Server, jest niekorzystne, dlatego powinieneś włączyć plan Wysoka wydajność.

Pozostawienie planu Zrównoważony może prowadzić do problemu *parkowania rdzeni* (ang. *core parking*). Polega to na tym, że niektóre rdzenie procesora są wyłączane. Problem może też wystąpić niezależnie od niskiej wydajności związanej z planem zasilania ustawionym na Zrównoważony.

Powinieneś zmierzyć wykorzystanie poszczególnych procesorów. Aby zbadać efekt niewłaściwie dobranego planu zasilania, wykorzystaj bezpłatne narzędzie CPU-Z. Możesz je pobrać ze strony <http://www.cpuid.com/softwares/cpu-z.html>.

Okno widoczne po lewej stronie rysunku 10.2 pochodzi z serwera z włączonym planem Zrównoważony. Jak widać, mamy tu ograniczenie szybkości procesora o wydajności 2,8 GHz do poziomu 0,798 GHz, co oznacza spadek o prawie 70%. Okno widoczne po prawej stronie dotyczy tego samego serwera po włączeniu planu Wysoka wydajność. Widać, że rdzenie komputera działają teraz z maksymalną wydajnością.

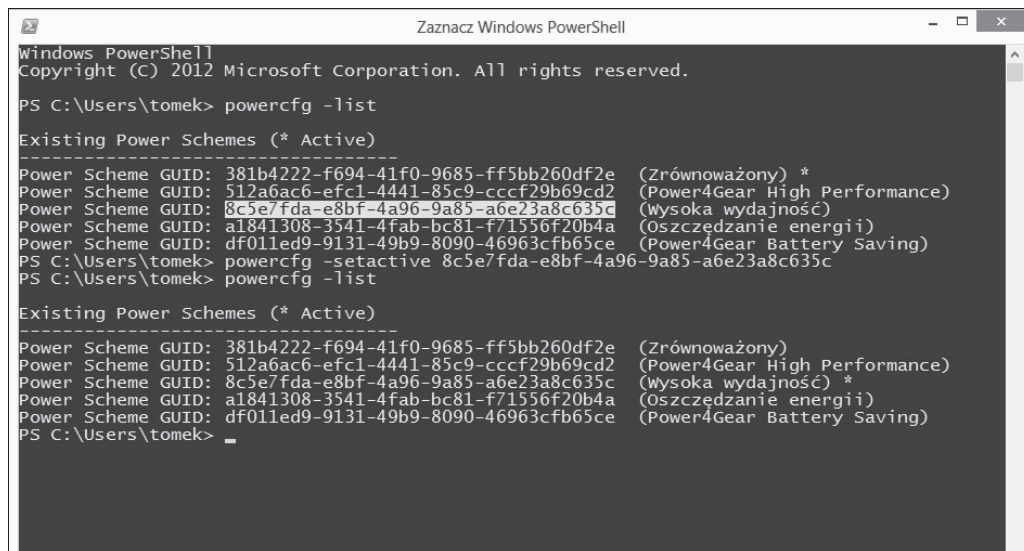
UWAGA Od systemu Windows Server 2012 parkowanie rdzeni jest domyślnie wyłączone, choć nadal trzeba zmienić ustawienia zasilania, aby w pełni wykorzystać możliwości procesora. Jednak od tej wersji nie trzeba się już przejmować parkowaniem rdzeni. Szczegółowe omówienie parkowania rdzeni i wykrywania tego zjawiska za pomocą monitora zasobów znajdziesz w artykule 281479 z bazy Microsoft Knowledge Base — <http://support.microsoft.com/kb/2814791>.



Rysunek 10.2. Wydajność procesora przy włączonych planach zasilania Zrównoważony (po lewej) i Wysoka wydajność (po prawej)

Aby włączyć odpowiednie ustawienia, najpierw należy wejść do BIOS-u serwera i umożliwić w nim zarządzanie zasilaniem przez system operacyjny. Jeśli nie zmienisz ustawień w BIOS-ie, system operacyjny nawet po wybraniu planu Wysoka wydajność nadal będzie działał zgodnie z planem Zrównoważony. Aby wprowadzić zmiany w BIOS-ie, należy ponownie uruchomić serwer i wejść do menu BIOS-a. Związane jest to z przestojem w pracy serwera.

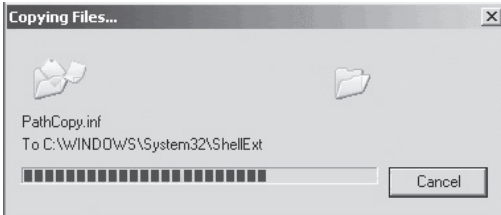
Po zmianie ustawień w BIOS-ie trzeba skonfigurować system operacyjny. W tym celu otwórz powłokę PowerShell na serwerze. Na rysunku 10.3 przedstawiono polecenia, które należy wpisać. Aby sprawdzić, który plan zasilania jest używany, wpisz instrukcję **powercfg -list**. Przy aktywnym planie widoczna jest gwiazdka (*). Aby zmienić plan, wprowadź polecenie **powercfg -setactive** i podaj identyfikator GUID planu zasilania, który chcesz zastosować.



Rysunek 10.3. Ustawianie planu zasilania w oknie powłoki PowerShell

Natychmiastowe inicjowanie plików bazy danych

Gdy kopiujesz duży plik na serwer plików lub między komputerami, czasem pojawia się okno, w którym widoczne są dokumenty przepływające między teczkami. W zależności od wielkości pliku ten proces może trwać przez pewien czas, a dopiero potem pojawiają się informacje o tym, jaka część pliku została już przesłana i kiedy proces kopiowania zostanie zakończony. Na rysunku 10.4 przedstawiono przykładową operację tego rodzaju.



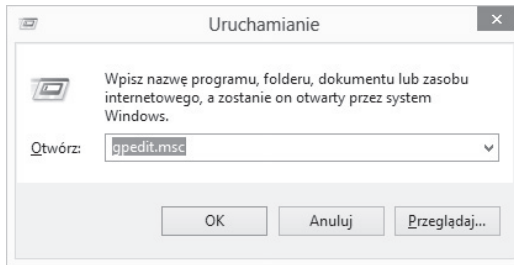
Rysunek 10.4. Kopiowanie dużego pliku

W czasie przed wyświetleniem informacji na temat kopiowanego pliku system operacyjny Windows zeruje docelowy plik. Proces polega na zainicjowaniu zerem każdego bitu danych w docelowym pliku. Dopiero potem rozpoczyna się kopiowanie. W systemie SQL Server można pominąć etap inicjowania zerami plików danych oraz kopii zapasowych bazy danych i dzienników. W tabeli 10.1 pokazano, że w ten sposób można poprawić wydajność systemu.

Tabela 10.1. Przykładowe wyniki operacji z natychmiastowym inicjowaniem plików bazy danych

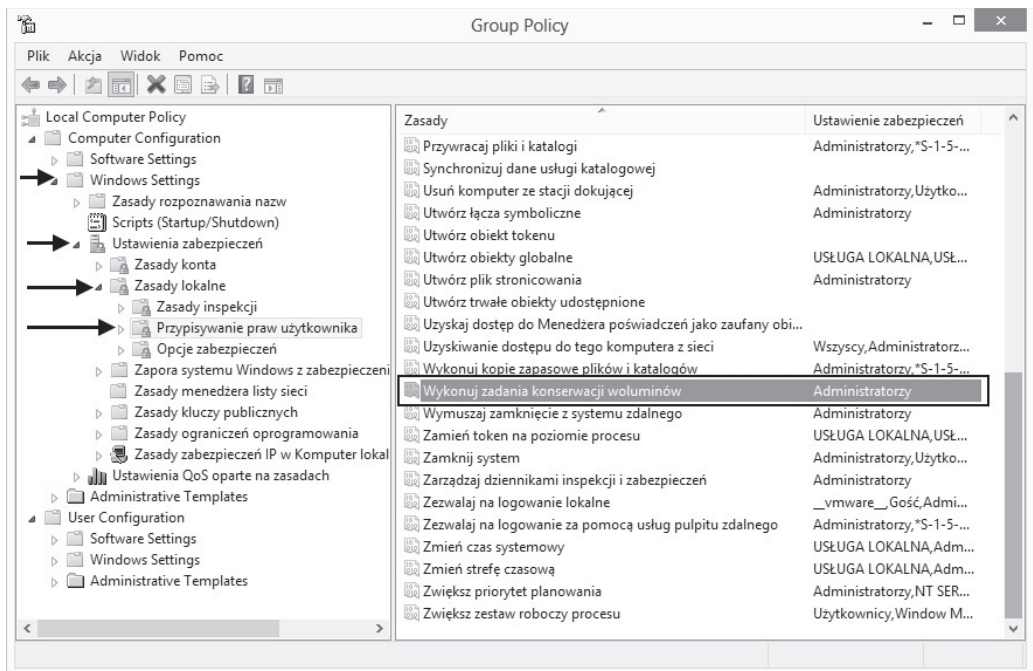
| Testy wydajności bez natychmiastowego inicjowania plików bazy danych | Wyniki (minuty:sekundy) |
|---|-------------------------|
| Tworzenie bazy danych (plik danych równy 20 gigabajtom) | 11:57 |
| Zmiana wielkości bazy o 10 gigabajtów | 5:59 |
| Tworzenie kopii zapasowej pustej bazy o pojemności 30 gigabajtów | 00:17 |
| Przywracanie pustej bazy o pojemności 30 gigabajtów | 17:46 |
| Przywracanie bazy o pojemności 30 gigabajtów z 12 gigabajtami danych | 25:07 |
| Tworzenie kopii zapasowej bazy o pojemności 30 gigabajtów z 12 gigabajtami danych | 17:21 |
| Testy wydajności z natychmiastowym inicjowaniem plików bazy danych | Wyniki (minuty:sekundy) |
| Tworzenie bazy danych (plik danych równy 20 gigabajtom) | 00:12 |
| Zmiana wielkości bazy o 10 gigabajtów | 00:00 |
| Tworzenie kopii zapasowej pustej bazy o pojemności 30 gigabajtów | 00:13 |
| Przywracanie pustej bazy o pojemności 30 gigabajtów | 00:07 |
| Przywracanie bazy o pojemności 30 gigabajtów z 12 gigabajtami danych | 6:50 |
| Tworzenie kopii zapasowej bazy o pojemności 30 gigabajtów z 12 gigabajtami danych | 13:43 |

Aby włączyć natychmiastowe inicjowanie plików, musisz ustawić konto usługowe (domeny lub lokalne) dla zasady *Wykonuj zadania konserwacji woluminów*. W tym celu otwórz program *gpedit.msc* w konsoli MMC. Najłatwiejszy sposób na wykonanie tej operacji to otwarcie okna *Uruchamianie* i wpisanie instrukcji *gpedit.msc*, co przedstawiono na rysunku 10.5.



Rysunek 10.5. Uruchamianie programu *gpedit.msc*

Rozwiń węzeł *Windows Settings/Ustawienia zabezpieczeń/Zasady lokalne/Przypisywanie praw użytkownika* przedstawiony na rysunku 10.6. Kliknij dwukrotnie *Wykonuj zadania konserwacji woluminów* i dodaj konto usługowe systemu SQL Server.



Rysunek 10.6. Dodaj odpowiednie konto do zasady *Wykonuj zadania konserwacji woluminów*

Aby sprawdzić, czy dla danej bazy włączone jest natychmiastowe inicjowanie plików, uruchom poniższą kwerendę i sprawdź zawartość dziennika błędów systemu SQL Server:

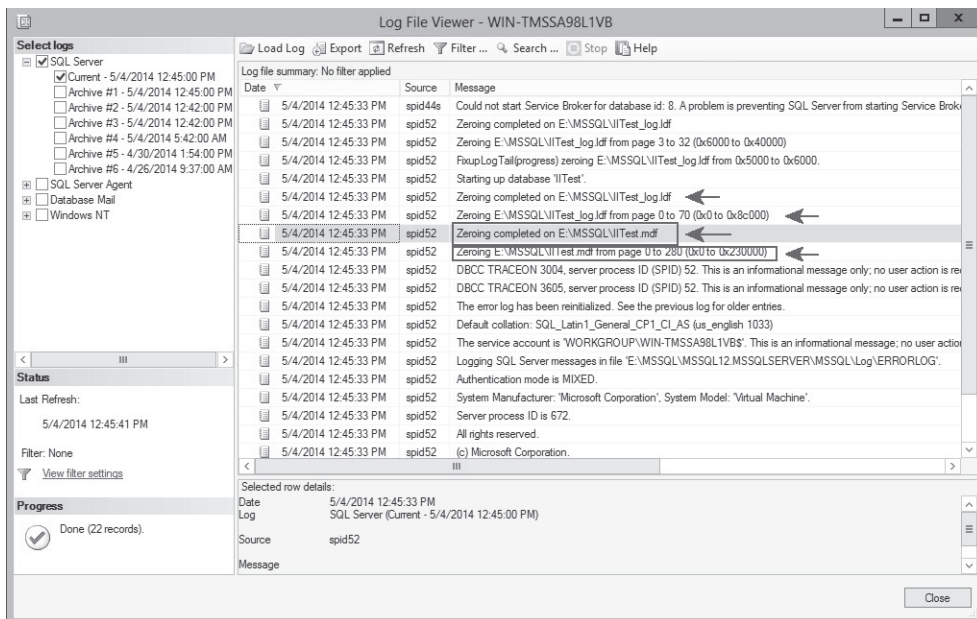
```
exec sp_cycle_errorlog
go
DBCC TRACEON(3605)
DBCC TRACEON(3004)
```

```

go
USE master
go
CREATE DATABASE IITest
DROP DATABASE IITest
go

```

Na rysunku 10.7 przedstawiono wyniki dla systemu, w którym natychmiastowe inicjowanie plików *nie jest włączone*. Opcja śledzenia 3605 włącza rejestrowanie poleceń DBCC w dzienniku błędów systemu SQL Server. Opcja śledzenia 3004 włącza rejestrowanie informacji i zmian w fizycznych plikach danych. Na rysunku 10.7 widać, że pliki *.mdf* i *.ldf* zostały wyzerowane w momencie tworzenia bazy danych. Gdyby natychmiastowe inicjowanie plików było włączone, wyzerowany zostałaby tylko plik *.ldf*. Dla pliku *.mdf* ta operacja zostałaby pominięta.



Rysunek 10.7. Zawartość dziennika z informacjami o zerowaniu plików

Możesz uruchomić przedstawiony kod dla istniejącej bazy danych, aby sprawdzić, czy natychmiastowe inicjowanie plików jest w niej włączone. Jeśli ten mechanizm nie jest włączony, musisz najpierw dodać konto do wspomnianej wcześniej zasady, a następnie ponownie uruchomić usługę systemu SQL Server.

UWAGA Jeśli zasady są przekazywane na lokalne serwery z nadrzędnego serwera domeny Active Directory, zastępują ustawienia lokalne. To oznacza, że jeśli chcesz wykorzystać zasady związane z usługą Active Directory, powinieneś porozmawiać z administratorem sieci o dodaniu odpowiedniej zasady do zasad tej usługi. Ponadto natychmiastowe inicjowanie plików nie współdziała z technologią TDE (ang. *Transparent Data Encryption*) i dziennikami transakcji. Może się zdarzyć, że bity z wcześniej usuniętych plików znajdują się w zapisanych w kopii zapasowej i przywróconych plikach *.mdf* i *.ndf*. Za pomocą narzędzi z obszaru informatyki śledczej można pobrać i odzyskać takie dane. Dlatego zawsze sprawdzaj, komu dajesz dostęp do plików danych.

Opcje śledzenia

W użytku zawsze jest wiele wersji systemu Microsoft SQL Server. Jego autorzy wciąż znajdują i naprawiają błędy oraz dodają nowe funkcje. Opcje śledzenia można dodać do systemu SQL Server za pomocą parametrów uruchomieniowych, np. `-T<numer opcji śledzenia>`, i rozdzielić przecinkami. Aby się dowiedzieć, jak ustawiać parametry uruchomieniowe, zajrzyj do rozdziału 2. Za pomocą opcji śledzenia można włączyć określone funkcje, które domyślnie są nieaktywne. Niektóre opcje śledzenia znacznie poprawiają wydajność. Ich opis zamieszczono w tabeli 10.2.

Tabela 10.2. Opcje śledzenia poprawiające wydajność

| Opcja śledzenia | Definicja |
|-----------------|---|
| -T1118 | Wyłącza alokację pojedynczych stron we wszystkich bazach z systemu SQL Server. Zwykle służy do zapobiegania współzawodnictwu o strony SGAM w bazie tempdb. |
| -T2371 | Umożliwia częstsze aktualizowanie statystyk w bazie danych. Domyślnie statystyki są aktualizowane po zmianie 20% zawartości bazy plus 500 wierszy. W małych tabelach to ustawienie sprawdza się dobrze. Jednak w dużych bazach zawierających miliony lub miliardy wierszy może spowodować, że statystyki szybko staną się nieaktualne. Ta opcja śledzenia powoduje obniżenie procentu wierszy, które muszą się zmienić, aby nastąpiła aktualizacja statystyk. |
| -T4199 | Dodaje wszystkie poprawki wprowadzone w procesorze kwerend w pakietach Service Pack i aktualizacjach wcześniejszych wersji systemu SQL Server. |
| -T3226 | Wyłącza nieustanne zapisywanie informacji o kopiach zapasowych w dzienniku błędów systemu SQL Server. Jeśli w danym egzemplarzu systemu działa od kilkudziesięciu do kilkuset baz danych, zobaczysz komunikat na temat każdej z nich. Jeżeli bazy pracują w trybie pełnego przywracania i zapisywane są kopie zapasowe dzienników transakcji, dziennik błędów stanie się nieczytelny. |

Istnieją też inne opcje śledzenia, które można włączyć. Większość z nich ma ściśle określone zastosowania. Opcje T1222 i T1204 są używane do śledzenia i wykrywania zakleszczeń. Opcja T610 pozwala usprawnić masowe wczytywanie indeksów klastrowanych i nieklastrowanych. Jednak opcja ta może negatywnie wpływać na pracę systemu OLTP z pełnym rejestrowaniem operacji i zwykle jest stosowana w hurtowniach danych.

Każdą opcję śledzenia należy przetestować przed zastosowaniem w systemie produkcyjnym. Pozwala to sprawdzić ogólny wpływ danej opcji na system.

Definicja dobrej wydajności

Podstawowe pytanie, które każdy administrator baz danych musi sobie zadać przed przystąpieniem do usprawniania systemu, brzmi: „Czy dany system pracuje obecnie z dobrą wydajnością?”. Bez określonego poziomu docelowego lub punktu odniesienia nie da się tego stwierdzić. Aby zebrać informacje potrzebne do udzielenia odpowiedzi na przedstawione pytanie, należy przygotować plan działań, oszacować ilość potrzebnych zasobów, przeprowadzić testy i monitorować pracę systemu. Cały ten proces można podzielić na trzy etapy.

1. Zaczynij od określenia krytycznego docelowego poziomu obciążenia procesora, pamięci i procesów wejścia-wyjścia.

2. Następnie ustal punkt odniesienia.
3. W ostatnim kroku, przed wdrożeniem systemu, obserwuj najważniejsze wskaźniki.

Wyobraź sobie wymagania dotyczące wydajności sklepu internetowego. Czas reakcji na działania użytkowników jest bardzo ważny, aby zachęcić ich do kontynuowania zakupów. Dlatego przy tworzeniu bazy takiego sklepu zwykle warto precyzyjnie zdefiniować czas reakcji na najważniejsze kwerendy. Menedżerowie mogą też zażądać, aby przetwarzanie dowolnej kwerendy zajmowało nie więcej niż 2 – 3 sekundy. Na innym serwerze bazodanowym, który generuje raporty z liczbą sztuk produktów w magazynie, można oczekiwać, że kwerendy będą potrzebowały czasu na pobranie potrzebnych informacji. W tej sytuacji akceptowalny jest czas reakcji na poziomie kilku minut. Nadal jednak może się okazać, że niektóre kwerendy powinny działać znacznie szybciej. W jeszcze innej bazie najważniejszym kryterium wydajności może być czas tworzenia kopii zapasowej bazy danych lub czas wczytywania albo przekazywania danych.

Po ustaleniu najważniejszych poziomów docelowych trzeba zmierzyć używany system, aby uzyskać punkt odniesienia. Techniki monitorowania pracy systemu SQL Server opisujemy szczegółowo w rozdziale 12. tej książki.

Koncentracja na tym, co najważniejsze

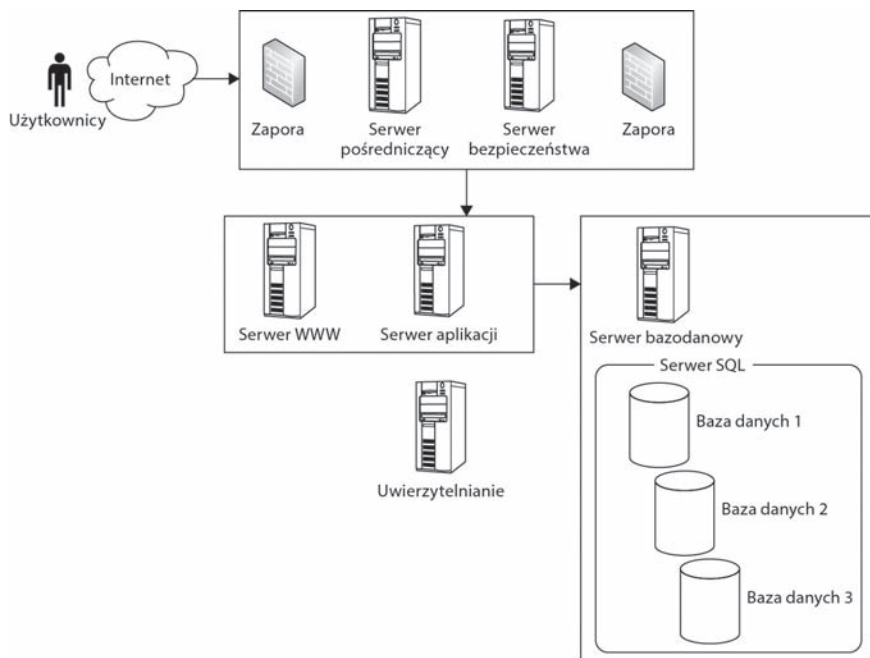
Ostatni ważny aspekt w obszarze wydajności to koncentracja na tym, co istotne — na uzyskaniu wydajności oczekiwanej przez użytkowników. Musisz ustalić, co masz zmierzyć, jak to zrobić, a także jakie błędy pomiaru mogą występować.

Pomyśl o typowym systemie. Odczucia użytkownika końcowego zależą od wydajności wielu elementów: od komputera danej osoby, przez wiele warstw pośrednich, po serwer bazodanowy (i z powrotem). Ponieważ ta książka dotyczy administratorów baz danych i systemu SQL Server 2014, możesz skoncentrować się na pomiarze wydajności tego systemu. Warto jednak zrozumieć ogólny kontekst, w jakim pracuje administrator baz danych, a także poznać narzędzia i miary, które mogą być przydatne w używanym systemie.

Na rysunku 10.8 przedstawiono schematyczny diagram typowej architektury opartej na sieci. Omówione tu rozwiązanie jest typowe dla klientów korporacyjnych używających architektury WSSRA (ang. *Windows Server System Reference Architecture*) do wdrażania rozbudowanych rozwiązań. Możliwe, że wielu czytających to administratorów baz danych po raz pierwszy widzi taki schemat i ma okazję zobaczyć, jakie miejsce w ogólnej architekturze zajmuje interesujący ich fragment (czyli baza danych).

W trakcie analizowania diagramu infrastruktury określ, na czym *powinna* polegać rola administratora bazy danych, a także na czym *polega* rola dobrego członka zespołu. W trakcie rozwiązywania problemów nie należy przyjmować podejścia „to nie moja sprawa”. Aby zapewnić właściwą pracę systemu SQL Server, trzeba zrozumieć, w jaki sposób aplikacje użytkowników, konfiguracja sieci i zabezpieczenia wpływają na ten system lub korzystają z niego.

Gdy użytkownik dzwoni do działu pomocy technicznej i narzeka na niską wydajność, liczba potencjalnych winowajców jest duża. Dlatego ustalenie, który fragment skomplikowanej architektury systemu jest przyczyną problemu, może zająć sporo czasu. Niestety, duże, złożone systemy wymagają wielu pracowników pomocy technicznej koncentrujących się na konkretnych częściach architektury. Przykładowo za zapory i serwer bezpieczeństwa odpowiada zespół ds. sieci, serwerem WWW i serwerem aplikacji zajmuje się zespół ds. aplikacji, o uwierzytelnianie



Rysunek 10.8. Diagram architektury WSSRA

dba zespół ds. systemu Windows, a systemem SQL Server opiekuje się zespół ds. administrowania bazami danych. Proces określania, kto w dużej lub małej firmie odpowiada za poszczególne zadania, polega na rozmowie z wszystkimi zespołami, co pomaga w całościowym rozwiązywaniu problemów.

Co administrator programista powinien wiedzieć na temat wydajności?

Dobra wydajność wynika z solidnych fundamentów, na których można zbudować resztę aplikacji. W bazach z systemu SQL Server tą podstawą jest dobrze zaprojektowany schemat bazy danych. Opisanie dalej zasady dostrajania wydajności są bardziej skomplikowane niż tradycyjne wskazówki w rodzaju „znormalizuj bazę do n-tej postaci normalnej”. Przedstawione tu porady wymagają dobrego zrozumienia sposobu korzystania z systemu, w tym wzorców użytkownika, instrukcji w SQL-u i danych. Optymalny schemat dla systemu OLTP może gorzej sprawdzać się w systemach DSS (ang. *Decision Support System*) lub DW (ang. *Data Warehousing*).

Użytkownicy

Najpierw musisz ustalić, kto będzie korzystał z systemu. Określ liczbę użytkowników i jak często jednocześnie korzystają oni z systemu. Ponadto ustal szczytowy poziom obciążenia i operacje wykonywane przez użytkowników. Użytkowników zwykle można przypisać do różnych grup na podstawie stanowisk lub stosowanych funkcji. W systemach obsługujących sklep internetowy grupami użytkowników mogą być: przeglądający produkty, kupujący, sprawdzający stan zamówienia, potrzebujący pomocy i inni. W systemach do analizy sprzedaży główną grupą

użytkowników są analitycy wczytujący dane za pomocą narzędzi do generowania raportów (np. programów PerformancePoint Server, Power View lub Excel) albo tworzący raporty dla zespołów sprzedawców. W systemie obsługującym sklep internetowy można utworzyć procesy pracy w bazie OLTP zoptymalizowane pod kątem mniejszej liczby szybszych operacji odczytu, aktualizacji i zapisu, natomiast w systemie analitycznym przydatne będą procesy pracy w bazie DSS zoptymalizowane pod kątem rozbudowanych kwerend potrzebnych przy generowaniu raportów.

Instrukcje w języku SQL

Po ustaleniu różnych grup użytkowników należy określić wykonywane przez nie operacje, wywoływane instrukcje w języku SQL i częstotliwość ich wykonywania w ramach działań użytkowników. W sklepie internetowym przeglądarka może kierować żądanie do witryny, która zwraca stronę główną. Może to wymagać wykonania 20 – 30 różnych procedur składowanych lub instrukcji w języku SQL. Gdy użytkownik kliknie element na stronie głównej, będzie to wymagać wykonania innego zestawu procedur składowanych i zwrócenia danych z następnej strony. Na razie wszystkie opisywane operacje dotyczą tylko odczytu, jednak na stronach ASP.NET należy też uwzględnić stan sesji, który może być przechowywany w bazie z systemu SQL Server. W takiej sytuacji opisane wcześniej czynności wymagają wielu operacji zapisu.

Wzorce używania danych

Ostatnim omawianym aspektem są dane w bazie. Musisz określić łączną ilość danych w każdej tabeli. Ustal, jak dane trafiają do tabel i jak się zmieniają. W systemie sklepu internetowego główne dane z witryny znajdują się w katalogu ze sprzedawanymi produktami. Produkty z katalogu mogą być pobierane bezpośrednio z witryn dostawców przez portal internetowy. Po początkowym wczytaniu dane można odświeżać, gdy dostawcy zmieniają oferowane produkty lub ich ceny. Ogólna ilość danych zmienia się w bardzo niewielkim zakresie, chyba że dodasz lub usuniesz produkty wybranych dostawców.

Zmieniają się natomiast (miejmy nadzieję, że szybko): liczba zarejestrowanych użytkowników, dane z systemu śledzenia zachowań użytkowników, liczba złożonych zamówień, liczba sprzedanych produktów i liczba dostarczonych zamówień. Oczywiście liczysz na to, że będziesz sprzedawał dużo produktów, co doprowadzi do codziennego wzrostu ilości rejestrowanych danych.

Dobra znajomość danych, ich rozmieszczenia i zmian pomoże wykryć kluczowe miejsca. Mogą nimi być np. punkty, w których dane są często pobierane, wstawiane lub aktualizowane. We wszystkich tych miejscach mogą wystąpić zatory obniżające wydajność systemu.

Schemat bazy danych

Wiedzę na temat wszystkich opisanych wcześniej elementów (użytkowników, instrukcji w języku SQL i danych) trzeba połączyć w całość, aby przygotować dobrze zaprojektowaną i wydajną aplikację. Jeśli podstawa, czyli schemat bazy danych, nie będzie solidna, wszystko, co zostanie na niej zbudowane, będzie niestabilne. Choć można wtedy uzyskać akceptowalne rozwiązanie, prawdopodobnie nie będzie ono optymalne.

W jaki sposób wszystkie wymienione informacje pomagają w dostrajaniu serwera? Dzięki znajomości używanych typów danych łatwiej będzie Ci zrozumieć, które zaawansowane funkcje można wykorzystać. Przykładowo kompresować można tylko dane przechowywane w wierszach.

Jeśli zatem korzystasz z typów LOB, np. `varbinary(max)`, typów tekstowych lub graficznych, nie będziesz mógł zastosować kompresji stron i wierszy. Podobnie nie można używać indeksów kolumnowych i technologii In-Memory OLTP, gdy niezbędne są ograniczenia.

Kolumny z ograniczeniami w postaci klucza zewnętrznego dobrze nadają się do tworzenia indeksów nieklastrowanych. Klucze używane do podziału na partycje zwykle powinny być statyczne, aby zapobiec przenoszeniu danych między partycjami. To proste przykłady, jednak pokazują, że znajomość danych przechowywanych w bazie pomaga zrozumieć, jak optymalnie skonfigurować system SQL Server.

Aby utworzyć maksymalnie wydajny projekt fizyczny, musisz ustalić kluczowe miejsca w danych. Jeśli projektujesz logiczny model danych, zwykle nie tworzysz go z myślą o optymalnej wydajności. Jednak na etapie projektowania modelu fizycznego trzeba uwzględnić zebrane informacje i dostosować projekt do wzorców dostępu do danych.

Co o wydajności powinien wiedzieć administrator produkcyjnej bazy danych?

Praca administratora produkcyjnej bazy danych znacznie różni się od zadań administratora programisty. Administrator produkcyjnych baz danych zarządza przejętym systemem, zaprojektowanym i zbudowanym przez kogoś innego (przy czym system może być nowy lub działać od jakiegoś czasu). Administrator może zmagać się z niską wydajnością starszych systemów, w których dawne aplikacje działają na przestarzałym sprzęcie. W takiej sytuacji zadanie nie polega na projektowaniu wydajnego systemu, ale na sprawieniu, aby przejęty system działał jak najlepiej przy obecnych ograniczeniach sprzętowych.

W tym scenariuszu najpierw trzeba ustalić możliwości sprzętu, zasoby sprzętowe potrzebne systemowi, a także oczekiwania użytkowników w zakresie czasu reakcji. Najważniejsze aspekty sprzętu to szybkość i typ procesora oraz wielkość jego pamięci podręcznej. Ponadto trzeba określić ilość dostępnej pamięci i szybkość magistrali. Należy też ustalić liczbę dysków używanych w operacjach wejścia-wyjścia, konfigurację tych dysków i liczbę kart sieciowych.

Następny krok wymaga określenia, jak wydajny ma być każdy komponent systemu. Czy między firmą a zespołem administratorów obowiązują umowy o gwarantowanym poziomie świadczenia usług dotyczące wydajności? Jeśli ustalone są wytyczne związane z wydajnością, to czy system spełnia je, przekracza, a może działa zbyt wolno? Ponadto zawsze powinniśmy znać obecny trend. Czy wydajność pozostaje bez zmian, poprawia się, a może — jak zdarza się najczęściej — powoli spada? Administratorzy produkcyjnych baz danych muszą zrozumieć wszystkie wymienione obszary, a także wiedzieć, jak wykrywać zatory i radzić sobie z nimi, aby system działał na oczekiwanym poziomie.

Poniżej znajdziesz listę narzędzi, z których administratorzy produkcyjnych baz danych korzystają przy wykonywaniu swoich zadań.

- **Menedżer zadań.** Narzędzie pozwala szybko zapoznać się z ogólnym obrazem wydajności serwera i wykorzystania zasobów.
- **Monitor wydajności systemu (w systemach Windows 2012 i Windows 8 używany jest Monitor wydajności; inna nazwa to Perfmon).** Zapewnia szczegółowe informacje o wydajności serwera z systemem Windows i liczniki specyficzne dla poszczególnych egzemplarzy systemu SQL Server.

- **Baza MDW systemu SQL Server.** Baza MDW to relacyjna baza danych rejestrująca i przechowująca informacje z monitora wydajności i kolektorów danych. Administrator może pobrać te dane, gdy rozwiązuje problemy systemowe. W systemie SQL Server 2014 można też profilować dane pod kątem ich wykorzystania w technologii In-Memory OLTP.
- **SQL Server Management Studio (SSMS).** Umożliwia analizowanie długich transakcji oraz wykrywanie i naprawianie zatorów. SSMS umożliwia administratorom wywoływanie kwerend kierowanych do widoków DMV i stosowanie zdarzeń rozszerzonych przy zbieraniu danych.
- **Widoki DMV.** Są to obiekty systemowe zawierające informacje o stanie serwera, pomocne przy diagnozowaniu problemów i monitorowaniu stanu systemu SQL Server.
- **Zdarzenia rozszerzone (ang. *extended events*).** Jest to prosty system monitorowania rejestrujący dane na temat wydajności systemu SQL Server. Dane z tego systemu można wyświetlać za pomocą interfejsu użytkownika sesji wprowadzonego w wersji SQL Server 2012. W wersji SQL Server 2014 dodano nowe zdarzenia rozszerzone pomocne przy monitorowaniu systemu i rozwiązywaniu problemów.

Szczegółowe omówienie tych narzędzi zamieszczono w rozdziale 13., „Dostrajanie wydajności kodu w języku T-SQL”.

UWAGA Po wprowadzeniu w wersji SQL Server 2014 narzędzia Transaction Performance Analysis Overview korzystanie z baz MDW zostało znacznie usprawnione. Teraz można zainstalować kolektory w systemach SQL Server 2008 i nowszych, a następnie zbierać dane w egzemplarzu systemu SQL Server 2014, aby ustalić tabele i procedury, które można wykorzystać w technologii In-Memory OLTP. Więcej informacji na ten temat znajdziesz w rozdziale 9., „Technologia In-Memory OLTP”.

Optymalizowanie sprzętu serwera

Pozostała część tego rozdziału dotyczy optymalizowania serwera. Dowiesz się, jak dobrać sprzęt i skonfigurować system operacyjny, aby zapewnić systemowi SQL Server najlepsze środowisko pracy. Przy omawianiu optymalizacji i wydajności zawsze powinieneś pamiętać o trzech podstawowych zasobach:

- procesorze,
- pamięci,
- mechanizmach wejścia-wyjścia.

Zacznijmy od procesora. Możliwości w tym zakresie są dość ograniczone — trzeba określić tylko liczbę i rodzaj procesorów. W tym fragmencie rozdziału koncentrujemy się na różnych cechach procesorów, co pomoże w zakupie odpowiedniego sprzętu.

W zakresie pamięci dostępnych jest znacznie więcej możliwości. Ponadto dużo łatwiej dodać pamięć RAM do systemu lub ją usunąć, niż zmienić liczbę i typ procesorów serwera. Gdy konfigurujesz serwer, powinieneś oszacować, ile pamięci możesz potrzebować. Określ też dostępne możliwości. Omów je z wybranym sprzedawcą. Ponieważ system SQL Server 2014 jest (i pozostanie) dostępny wyłącznie jako serwer 64-bitowy, powinieneś zainstalować go na komputerze 64-bitowym z systemem operacyjnym Windows Server 2012 R2 (też 64-bitowym), co opisujemy dalej w tym rozdziale.

Wydajność operacji wejścia-wyjścia jest prawdopodobnie najważniejszym aspektem konfiguracji serwera, ponieważ wszystkie operacje są wykonywane na dyskach. Cały kod uruchamiany w systemie operacyjnym, systemie SQL Server i innych aplikacjach początkowo znajduje się w plikach na dysku. Wszystkie dane używane w systemie SQL Server także są przechowywane na dysku — są na nim zapisane, potem są wczytywane do pamięci, a następnie ponownie zapisywane na dysku w celu utrwalenia zmian. Każda modyfikacja w bazie z systemu SQL Server jest zapisywana w pliku dziennika transakcji bazy danych, który także znajduje się na dysku. Wszystko to sprawia, że poprawne skonfigurowanie operacji wejścia-wyjścia to bardzo ważny aspekt każdego egzemplarza systemu SQL Server.

W kolejnych punktach szczegółowo omawiamy podstawowe zasoby. Jednak najpierw warto zastanowić się nad tym, jak te trzy zasoby zależą od siebie w kontekście wydajności. Możesz teraz zapisać w książce początkową wydajność procesora, pamięci i operacji wejścia-wyjścia, aby w przyszłości zobaczyć, jak zmieniła się wydajność każdego z tych elementów.

W systemie SQL Server do wersji 2014 nie można było w dużym stopniu zmienić ilości danych przetwarzanych w każdym cyklu procesora. Jednak obecnie natywnie skompilowane procedury składowane (opisane w rozdziale 9., „Technologia In-Memory OLTP”) pozwalają skompilować procedury składowane w języku T-SQL, aby zmniejszyć zbiór instrukcji potrzebnych w trakcie przetwarzania kodu.

Nie każdy kod w języku T-SQL nadaje się do wykorzystania w natywnie skompilowanych procedurach składowanych. Przykładowo skomplikowane kwerendy, ograniczenia i funkcje nie sprawdzają się dobrze w takich procedurach. Ponadto w omawianych procedurach można używać tylko typów danych technologii In-Memory OLTP. Jednak niezależnie od stosowania takich procedur warto pomyśleć o zakupie procesorów z dużą pamięcią podręczną i szybkością. Warto dodać więcej pamięci i zaprojektować podsystem przechowywania danych w taki sposób, aby zapewnić jego maksymalną wydajność w ramach ograniczeń dotyczących szybkości, pojemności i kosztów.

Wniosek z tego fragmentu jest taki, że najwolniejszy element sprzętu ogranicza pozostałe komponenty. Magnetyczne dyski fizyczne nigdy nie będą tak szybkie jak pamięć RAM. System SQL Server (niezależnie od tego, czy używana jest technologia In-Memory OLTP, czy tradycyjny język T-SQL) to aplikacja działająca w pamięci, co oznacza, że wszystkie zadania są wykonywane na danych wczytanych najpierw do pamięci. Gdy określasz wymagania sprzętowe dla systemu SQL Server, upewnij się, że zapewnisz mu wystarczającą ilość pamięci. Raczej nie zdarza się, aby administrator bazy danych narzekał na za dużą ilość pamięci. Z drugiej strony zbyt mała jej ilość często prowadzi do problemów z wydajnością.

Zarządzanie sprzętem

W większości małych i średnich serwerów bazodanowych często włącza się w BIOS-ie technologię Hyper-Threading (szczegółowe informacje na ten temat znajdziesz w dokumentacji serwera). W punkcie „Technologia Hyper-Threading”, dalej w tym rozdziale opisano, jak ustalić, czy ta technologia umożliwi poprawienie wydajności systemu. Po podjęciu decyzji dotyczącej ustawień technologii Hyper-Threading należy fizycznie zainstalować pamięć RAM, urządzenia wejścia-wyjścia (np. karty sieciowe) i adaptery dysków SCSI (ang. *Small Computer Systems Interface*) lub SATA (ang. *Serial Advanced Technology Attachment*). Dodatkowe informacje o zalecanych ustawieniach znajdziesz w dokumentacji udostępnionej przez producentów.

W prawie wszystkich systemach dostępne jest oprogramowanie pomocne przy konfigurowaniu i obsłudze sprzętu oraz zarządzaniu nim. Większość producentów sprzętu udostępnia własne oprogramowanie tego rodzaju, oferujące wiele możliwości i opcji. Są to np. narzędzia iLO firmy Hewlett Packard, RSA firmy IBM i DRAC firmy Dell.

W dużych systemach korporacyjnych, takich jak HP Superdome 2, NEC Express5800 lub SGI Altix UV, konfigurowanie sprzętu serwera odbywa się na zupełnie innym poziomie. W takich systemach używany jest *procesor serwisowy* (ang. *management processor*). Procesor serwisowy i powiązane z nim oprogramowanie pozwalają kontrolować sprzęt: od ładowania partycji, przez konfigurowanie różnych partycji i zmianę układu pamięci, po zarządzanie zasilaniem różnych komponentów. Za wykonywanie wszystkich tych zadań odpowiada właśnie procesor serwisowy.

Zadania wykonywane przy zarządzaniu różnymi dużymi systemami są podobne, jednak każdy producent stosuje inny rodzaj interfejsu: od witryn opartych na Javie w systemach SGI po telnetowy interfejs w wierszu poleceń w rozwiązaniach firm HP i NEC.

UWAGA Gdy rozważasz zakup nowego sprzętu, powinieneś zacząć od przejrzania katalogu Windows Server Catalog (<http://www.windowsservercatalog.com>). Jeśli dany sprzęt nie jest wymieniony w tym katalogu, nie współdziała z systemem Windows Server 2012 i nie będzie obsługiwał także systemu SQL Server.

Procesor

System SQL Server 2014 działa w innym środowisku niż wcześniejsze jego wersje. Gdy wprowadzono na rynek system SQL Server 2000, duże serwery używane do uruchamiania takich systemów miały od 4 do 8 procesorów. Obecnie system SQL Server 2014 może działać na największych serwerach mających do 64 procesorów i do 320 rdzeni. Ponadto system SQL Server 2014 współdziała z maszynami mającymi do 4 terabajtów pamięci RAM pracującymi pod kontrolą systemu Windows Server 2012 R2 Standard Edition. Jedyny powód, dla którego warto wybrać system operacyjny Windows Server 2012 R2 Data Center, to możliwość obsługi więcej niż dwóch gniazd na procesor. System SQL Server 2014 współdziała z tylko jedną architekturą procesorów — x64 (czyli z architekturą 64-bitową).

Architektura x64

Architektura x64 została wprowadzona przez firmę AMD. Intel zaimplementował tę architekturę pod nazwą EM64T. Jest ona zgodna z kodem maszynowym dla procesorów x86 i obsługuje mikrokod 64-bitowy. Platforma z architekturą x64 obsługuje system SQL Server 2014 i system operacyjny Windows Server 2012 R2 oraz pamięć RAM przekraczającą 4 gigabajty (aż do 4 terabajtów natywnie adresowalnej pamięci) i do 64 fizycznych procesorów z 320 logicznymi rdzeniami.

UWAGA Platforma z architekturą x64 jako jedyna obsługuje system SQL Server 2014. Procesory z rodzin Itanium 64 (IA64) i x32 (x86-32) nie współdziałają z tym systemem. Dostępne są jednak różne edycje i wersje systemów SQL Server 2008 i 2008 R2 obsługiwane przez procesory z wymienionych rodzin.

Pamięć podręczna

W nowoczesnych procesorach potrzebna jest pamięć podręczna, ponieważ działają one z szybkością 2 – 3 GHz. Choć szybkość głównej pamięci RAM jest coraz większa, wciąż jest za niska w porównaniu z prędkością procesora. Aby ograniczyć skutki tego problemu, projektanci procesorów dodali kilka poziomów pamięci flash, co pozwala na przechowywanie używanych ostatnio danych w małych, szybkich blokach pamięci podręcznej. Gdy zatem dane te są ponownie potrzebne, można uzyskać do nich szybki dostęp.

Pamięć podręczna w nowoczesnych procesorach ma zwykle kilka warstw: L1, L2 i L3. Każda kolejna warstwa jest bardziej oddalona od rdzenia procesora i większa (a zarazem wolniejsza). Ostatnią warstwą jest główna pamięć RAM. Niektóre warstwy pamięci podręcznej są przeznaczone do użytku ogólnego i przechowują kopie danych z dowolnej pamięci (np. z L2 lub L3).

Warto przyrzeć się wydajności pamięci podręcznej w kontekście innych rodzajów pamięci. Pamięć systemowa działa średnio z opóźnieniem 50 – 100 nanosekund i ma pojemność od 16 do 1024 gigabajtów. Dyski SSD cechują się opóźnieniem 30 – 100 mikrosekund i pojemnością 50 – 1024 gigabajtów. Profesjonalne dyski twarde mają opóźnienie na poziomie 2 – 50 milisekund i pojemność od 80 do 2048 gigabajtów.

Wydajność systemu SQL Server w dużym stopniu zależy od pojemności pamięci podręcznej. Producenci procesorów udostępniają wiele modeli z pamięciami L2 i L3 o różnej pojemności. Uzyskanie wysokiej wydajności oferowanej przez pamięć podręczną jest kosztowne. Dlatego procesory o dużej ilości pamięci podręcznej są drogie. Do serwera powinienś zakupić możliwie najszybszy procesor o jak największej pojemności pamięci podręcznej. Jeśli musisz wybrać kompromisowe rozwiązanie, pamiętaj, że zawsze łatwiej i taniej dokupić pamięć RAM, niż zaktualizować procesor.

UWAGA W 2013 roku Intel udostępnił pierwszy chipset z rozbudowaną pamięcią podręczną L4 o pojemności 128 megabajtów. Ten chipset jest obecnie dostępny tylko w procesorach na urządzenia przenośne. Pojawiły się jednak plotki, że procesor Broadwell (następca procesora Haswell) też będzie miał pamięć podręczną L4. W czasie, gdy powstawała ta książka, procesor Broadwell nie był jeszcze dostępny. Zgodnie z doniesieniami pamięć podręczna L4 działa z opóźnieniem 50 – 60 nanosekund.

Technologia Hyper-Threading

Hyper-Threading to zastrzeżona technologia firmy Intel, która logicznie duplikuje wybrane fragmenty fizycznego rdzenia procesora, aby zwiększyć współbieżność obliczeń. W efekcie dla każdego rdzenia fizycznego w systemie operacyjnym pojawiają się dwa rdzenie logiczne. Choć system szereguje do wykonania w procesorze wiele wątków, współużytkowanie zasobów może powodować, że niektóre wątki muszą oczekiwać na zakończenie pracy przez inne.

W kontekście technologii Hyper-Threading trzeba zastanowić się nad tylko jedną kwestią: czy ma być włączona, czy nie. Technologię Hyper-Threading należy domyślnie włączać. Wyłączać ją trzeba tylko wtedy, gdy powoduje spadek wydajności.

Ważnym czynnikiem związanym z technologią Hyper-Threading jest ustalenie maksymalnego teoretycznego wzrostu wydajności, jaki umożliwia. Według dokumentacji Intela maksymalna teoretyczna poprawa wydajności wynosi 30%. Zauważ więc, że ta technologia może w najlepszym razie zwiększyć maksymalną wydajność tylko 1,3 razy w porównaniu z sytuacją, gdy nie jest stosowana. W praktyce ten wzrost może być bliższy 1,1 – 1,15 razy.

W niektórych sytuacjach technologia Hyper-Threading (przynajmniej teoretycznie) nie przynosi żadnych korzyści. Przykładowo w procesach pracy, w których kod działa w krótkiej pętli i wszystkie dane są przechowywane w pamięci podręcznej, omawiana technologia nie jest pomocna, ponieważ używany jest tylko jeden silnik wykonania. Ten scenariusz może prowadzić do spadku wydajności, ponieważ system operacyjny próbuje wtedy przydzielić zadania procesorowi, który fizycznie nie istnieje.

Aby zmierzyć przyrost wydajności, powinieneś przeprowadzić testy porównawcze. Wykonaj następujące kroki.

1. Najpierw sprawdź w dokumentacji systemu, jak wyłączyć technologię Hyper-Threading i zrób to.
2. Potem przeprowadź kilkakrotnie test, aby otrzymać średni czas wykonania.
3. Następnie ponownie włącz technologię Hyper-Threading.
4. Przeprowadź test tyle samo razy, co wcześniej, i porównaj wyniki. Celem jest ustalenie, o ile krótszy jest średni czas wykonania testu z włączoną technologią Hyper-Threading. Oblicz różnicę między czasem z obu testów i na podstawie uzyskanego wyniku ustal, czy warto stosować omawianą technologię.

Pojęcia związane z systemami wielordzeniowymi

W tym miejscu warto zapoznać się z definicjami wybranych pojęć, aby uniknąć wątpliwości podczas poznawania systemów wielordzeniowych.

- *Gniazdo* to fizyczne gniazdo, w które wkładany jest procesor. Przed pojawieniem się systemów wielordzeniowych jednemu gniazdu odpowiadała jedna jednostka wykonawcza.
- *Rdzeń* to *jednostka wykonawcza* (wcześniej za taką jednostkę uznawano procesor). W procesorach wielordzeniowych na gniazdo przypadają dwa rdzenie lub większa ich liczba.
- *Wątek* w tym kontekście nie jest tym samym, co wątek tworzony w programie lub wątek systemu operacyjnego. Tu wątek ma znaczenie tylko w kontekście technologii Hyper-Threading. Taki wątek nie jest nową jednostką wykonania; bardziej przypomina nowy potok istniejącej jednostki wykonania. Więcej informacji na temat tego rozwiązania znajdziesz w poprzednim punkcie, „Technologia Hyper-Threading”.

Dla każdego egzemplarza systemu SQL Server 2014 trzeba wykupić licencję przynajmniej na 4 rdzenie. Niezależnie od tego, czy będą one fizyczne, czy wirtualne, każdy egzemplarz systemu SQL Server 2014 powinien mieć dostęp do 4 rdzeni. W kontekście technologii Hyper-Threading istotne jest to, że użytkownik płaci tylko za fizyczne rdzenie używane przez fizyczny egzemplarz systemu SQL Server. Nie trzeba ponosić opłat za dodatkowe rdzenie logiczne udostępniane za pomocą tej technologii.

Architektura NUMA

Nazwa NUMA to akronim *non-uniform memory access* (czyli niejednorodny dostęp do pamięci). Architektura ta często jest nazywana ccNUMA (ang. *cache-coherent NUMA*), czyli NUMA spójna z pamięcią podręczną. Główne różnice między starszym systemem SMP a architekturą NUMA dotyczą miejsca podłączenia pamięci i uporządkowania procesorów na magistrali systemowej.

W systemie SMP pamięć jest podłączona do wszystkich procesorów symetrycznie za pomocą współużytkowanej magistrali. W architekturze NUMA każda grupa procesorów ma własną pulę lokalnej pamięci. Zaletą tego modelu jest to, że procesory przy dostępie do pamięci nie muszą ponosić kosztów wychodzenia poza pamięć lokalną (pod warunkiem, że potrzebne dane się w niej znajdują). Gdy szukane dane są zapisane w puli pamięci innego węzła architektury NUMA, koszt dostępu do nich jest nieco wyższy niż w systemie SMP. Dlatego jednym z celów przy stosowaniu architektury NUMA jest zmaksymalizowanie ilości danych pobieranych z pamięci lokalnej i zminimalizowanie dostępu do danych z innych węzłów.

W architekturze NUMA zwykle stosuje się dwa gniazda na węzeł i tworzy się tyle węzłów, ile pozwala na to system.

Mniejsze systemy z węzłami, w których liczba gniazd jest wielokrotnością dwójki, można zwykle modyfikować. Dzięki temu serwery mają duże możliwości w zakresie konfigurowania i skalowania. Przykładowo firma może zacząć od jednego węzła z dwoma gniazdami, a następnie rozbudować system do szesnastu węzłów po cztery gniazda każdy (w sumie daje to sześćdziesiąt cztery gniazda).

Procesory Intel Xeon i AMD Opteron wykorzystują odmienną architekturę dostępu do pamięci. W procesorach Intela używa się magistrali FSB (ang. *Front Side Bus*), w której gniazda są za pomocą magistrali podłączone przez zewnętrzny kontroler do pamięci. W efekcie wszystkie gniazda znajdują się w tej samej odległości od pamięci. W procesorach AMD w każdym gnieździe używany jest zintegrowany kontroler pamięci zapewniający dostęp do lokalnej pamięci i do pamięci innych gniazd. Do transmisji danych stosowana jest tu technologia HyperTransport. Taki niejednorodny dostęp do pamięci nazywany jest architekturą NUMA. Opóźnienie w dostępie do danych zależy tu od tego, w którym miejscu pamięci znajdują się dane zażądane przez określony rdzeń procesora.

Jeśli np. dane znajdują się w bezpośrednio podłączonym bloku pamięci, czas dostępu jest krótki. Jeżeli jednak dane znajdują się w zdalnym bloku pamięci w innym gnieździe, opóźnienie jest większe. Choć w architekturze Intela magistrala FSB zapewnia każdemu rdzeniowi procesora dostęp do pamięci oddalonej zawsze o identyczną odległość, współzawodnictwo o samą magistralę może prowadzić do spadku wydajności. Intel ogranicza ten problem za pomocą pamięci podręcznej o większej pojemności.

Pamięć

Innym podsystemem sprzętowym, który trzeba uwzględnić, jest pamięć. Ważna jest głównie pamięć serwera, w tym związane z nią problemy, dostępne opcje i jej wpływ na wydajność serwera. W następnym punkcie znajdziesz podstawowe wprowadzenie do pamięci systemu operacyjnego. Dalej szczegółowo omawiamy konfigurowanie serwera w obszarze pamięci.

Pamięć fizyczna

Pamięć fizyczna to pamięć RAM instalowana na serwerze. Prawdopodobnie znasz już pamięć DIMM (ang. *Dynamic Inline Memory Modules*) używaną w komputerach PC i serwerach. Jest to jedna z odmian pamięci fizycznej (pamięci RAM). Pojemność takiej pamięci mierzy się w megabajtach, gigabajtach lub — jeśli masz szczęście — w terabajtach, ponieważ najnowsze systemy operacyjne Windows Server 2012 R2 Datacenter i Standard obsługują do 4 terabajtów

pamięci RAM. W nowszych wersjach tego systemu operacyjnego ilość obsługiwanej pamięci będzie jeszcze większa, ponieważ klienci oczekują jeszcze wydajniejszych narzędzi do rozwiązywania coraz bardziej skomplikowanych problemów biznesowych.

Fizyczna przestrzeń adresowa

Fizyczna przestrzeń adresowa to zbiór adresów używany przez procesor do dostępu do elementów z magistrali. Dużą część tej przestrzeni zajmuje pamięć, natomiast inne fragmenty są zarezerwowane np. na odwzorowania buforów sprzętowych i specyficzną pamięć (taką jak np. pamięć RAM dla przetwarzania grafiki). W procesorach 32-bitowych wielkość przestrzeni adresowej była ograniczona do 4 gigabajtów. W przeznaczonych dla serwerów 32-bitowych procesorach Intel'a z technologią PAE (ang. *Physical Address Extension*) używana była 36-bitowa magistrala adresowa, co pozwalało na obsługę 64 gigabajtów adresów pamięci. Mogłoby się wydawać, że w procesorach 64-bitowych stosowana jest 64-bitowa magistrala adresowa, jednak ponieważ na razie nie są potrzebne systemy adresujące 18 eksabajtów pamięci (nie jest też możliwe zbudowanie tak dużego systemu), producenci ograniczyli magistralę adresową do 48 bitów, co wystarcza na adresy dla 256 terabajtów pamięci. Używana architektura umożliwia rozszerzenie magistrali adresowej do 52 bitów, co pozwoli na tworzenie systemów mających do 4 petabajtów pamięci.

Menedżer pamięci wirtualnej

Menedżer pamięci wirtualnej (ang. *Virtual Memory Manager* — VMM) to część systemu operacyjnego zarządzająca całą pamięcią fizyczną i udostępniającą ją wszystkim procesom, które jej potrzebują. Zadaniem menedżera VMM jest udostępnianie każdemu procesowi pamięci, gdy jej zażąda, przy czym fizyczna pamięć jest współużytkowana przez wszystkie procesy działające w danym momencie w systemie.

Menedżer VMM zarządza pamięcią wirtualną każdego procesu i — jeśli trzeba — przywraca powiązaną z nią pamięć fizyczną oraz umieszcza zapisane w niej dane w pliku stronicowania, aby nie zostały utracone. Gdy proces ponownie będzie potrzebował tej pamięci, menedżer VMM przywróci dane z pliku stronicowania, znajdzie wolną stronę pamięci (albo na liście wolnych stron, albo w innym procesie), umieści na niej dane z pliku stronicowania i powiąże nową stronę z wirtualną przestrzenią adresową procesu. Związane z tym opóźnienie lub przerwanie to *błąd strony* (ang. *page fault*). Aby ustalić, czy system SQL Server lub inny proces nie powodują nadmiernego stronicowania, sprawdź wartość licznika *Proces: Błędy stron/s* dla procesu systemu SQL Server. Szczegółowe informacje na ten temat zamieszczono w punkcie „Błędy strony”, dalej w tym rozdziale.

W systemach z wystarczającą ilością pamięci RAM, w których każdemu procesowi można przydzielić całą potrzebną pamięć, menedżer VMM nie ma wiele pracy. Musi tylko udostępniać pamięć i wykonywać operacje porządkujące, gdy proces już jej nie potrzebuje. W systemach, w których brakuje pamięci RAM, sytuacja jest bardziej skomplikowana. Menedżer VMM musi wykonać określone zadania, aby w odpowiednim czasie udostępnić każdemu procesowi potrzebną pamięć. Używany jest do tego plik stronicowania. Zawiera on dane ze stron, których proces nie używa lub które według menedżera VMM można usunąć z procesu.

Plik stronicowania

Plik stronicowania (ang. *page file*) to przechowywany na dysku plik, używany przez komputer do zwiększenia ilości fizycznej pamięci wykorzystywanej w pamięci wirtualnej. Gdy pamięć używana przez wszystkie istniejące procesy przekracza ilość dostępnej pamięci RAM, system operacyjny Windows przenosi strony z wirtualnych przestrzeni adresowych do pliku stronicowania przechowywanego na fizycznym dysku. To pozwala zwolnić pamięć RAM dla innych zadań. Przeniesione strony są przechowywane w plikach stronicowania w katalogu głównym partycji. W każdej partycji może znajdować się tylko jeden taki plik.

Na serwerze z systemem SQL Server należy postarać się, aby ten system używał tylko dostępnej pamięci fizycznej. Sam system SQL Server w dużym stopniu dba o to, by nie zajmować zbyt dużej ilości pamięci, i próbuje nie przekraczać limitu dostępnej pamięci fizycznej.

Dzięki temu w większości sytuacji plik stronicowania jest potrzebny tylko w małym stopniu. Jednak wielu użytkowników zadaje pytanie: „Jaka jest zalecana wielkość pliku stronicowania?”. Oczywiście odpowiedź na to pytanie brzmi: „To zależy”. Wpływa na to ilość zainstalowanej pamięci RAM i zapotrzebowanie na pamięć wirtualną ze strony procesów innych niż procesy systemu SQL Server. Zwykle plik stronicowania powinien mieć pojemność 1,5 – 2 razy większą niż ilość pamięci RAM serwera.

W rozbudowanych systemach o dużej ilości pamięci RAM (powyżej 128 gigabajtów) czasem nie da się skorzystać z tej porady z powodu braku miejsca na dysku. Oto wskazówki, które mogą okazać się przydatne w takich sytuacjach.

- Utwórz na dysku systemowym plik stronicowania o pojemności 8 gigabajtów.
- Upewnij się, że parametry uruchomieniowe systemu operacyjnego Windows powodują zapis zrzutu jądra przy wystąpieniu awarii.

UWAGA Aby się dowiedzieć, jak skonfigurować te parametry, zapoznaj się z artykułem <http://support.microsoft.com/kb/307973> z witryny z pomocą techniczną Microsoftu.

- Opcjonalnie możesz skonfigurować kilka plików stronicowania (na woluminach innych niż wolumin systemowy), udostępnianych systemowi operacyjnemu wtedy, gdy potrzebna jest większa pojemność tego pliku.

W niektórych sytuacjach system SQL Server i system operacyjny nie współdziałają dobrze w zakresie współużytkowania dostępnej pamięci. Pojawiają się wtedy ostrzeżenia systemowe dotyczące małej ilości pamięci wirtualnej. Wtedy najlepiej dodać do serwera więcej pamięci RAM, zmienić konfigurację systemu SQL Server, aby zużywał mniej pamięci, lub zwiększyć rozmiar pliku stronicowania. Czasem lepiej tak ustawić system SQL Server, aby nie wykroczył poza dostępną pamięć fizyczną, niż powiększać plik stronicowania. Ograniczenie stronicowania zawsze prowadzi do wyższej wydajności. Jeśli stronicowanie jest nieuniknione, to aby uzyskać optymalną wydajność, należy umieścić plik stronicowania na szybkim i rzadko używanym dysku. Dysk ten należy okresowo defragmentować, aby plik stronicowania zajmował ciągły obszar pamięci. Zmniejsza to konieczność przesuwania głowicy dysku i zwiększa wydajność. Do pomiaru wykorzystania pliku stronicowania służy wskaźnik *Plik stronicowania: Użycie%* w monitorze wydajności systemu Windows. Jego wartość nie powinna przekraczać 70%.

Błędy strony

Błędy strony zwykle sprawiają problemy w systemie SQL Server, jednak nie wszystkie błędy tego rodzaju są takie same. Niektórych nie da się uniknąć, inne mają niewielki wpływ na wydajność, a jeszcze inne mogą prowadzić do poważnego spadku wydajności — tych należy się wystrzeżać.

System SQL Server jest zaprojektowany tak, aby korzystać z takiej ilości pamięci, jaka fizycznie jest dostępna. Ma to chronić przed szkodliwymi błędami strony. Niestety, licznik błędów strony w monitorze wydajności nie informuje, czy skutki danego błędu są niewielkie, czy poważne. Dlatego nie da się określić wpływu danego błędu na wydajność.

Miękkie błędy strony

Najczęściej występującym typem omawianego problemu są *miękkie błędy strony*. Występują one wtedy, gdy potrzebna jest nowa strona pamięci. Za każdym razem, gdy system SQL Server chce zająć dodatkową pamięć, żąda od menedżera VMM nowej strony pamięci. Menedżer zgłasza wtedy miękki błąd strony, aby przenieść daną pamięć do wirtualnej przestrzeni adresowej systemu SQL Server. Dzieje się to, gdy SQL Server po raz pierwszy chce użyć danej strony, a nie wtedy, gdy po raz pierwszy jej zażąda. System SQL Server musi wywołać instrukcję `VirtualAlloc`, aby otrzymać stronę pamięci. Tego rodzaju błąd strony występuje tylko wtedy, gdy system SQL Server po raz pierwszy próbuje zapisać dane na określonej stronie.

Twarde błędy strony

Twarde błędy strony to te, których powinieneś unikać. Błędy te występują wtedy, gdy system SQL Server próbuje uzyskać dostęp do strony pamięci przeniesionej do pliku stronicowania. W takiej sytuacji menedżer VMM musi wykonać odpowiednie operacje, aby pobrać potrzebną stronę z pliku stronicowania z dysku, znaleźć pustą stronę pamięci, wczytać stronę z dysku, zapisać ją na nowej pustej stronie, a następnie powiązać nową stronę z przestrzenią adresową systemu SQL Server. Przez cały ten czas wątek systemu SQL Server musi czekać. Dopiero po zastąpieniu brakującej strony pamięci przez menedżer VMM system SQL Server może kontynuować wykonywane zadania.

Operacje wejścia-wyjścia

Konfigurowanie operacji wejścia-wyjścia to zbyt obszerny temat, aby można go było opisać w jednym rozdziale. To zagadnienie zasługuje na odrębną książkę. W tym podrozdziale omawiamy wybrane z dostępnych opcji wejścia-wyjścia, a także przedstawiamy kilka scenariuszy. Pomoże Ci to zrozumieć, jak podejmować właściwe decyzje w trakcie konfigurowania systemu pamięci.

Operacje wejścia-wyjścia dotyczą zarówno sieci, jak i dysków twardych. W systemie SQL Server w większości sytuacji istotne są przede wszystkim operacje wejścia-wyjścia na dyskach twardych, ponieważ to tam przechowywane są dane. Powinieneś jednak zrozumieć także skutki wolnego działania operacji wejścia-wyjścia w sieci, ponieważ mogą one powodować zatory i obniżyć wydajność systemu.

Konfigurowanie operacji wejścia-wyjścia dla systemu pamięci serwera to obszar, w którym masz prawdopodobnie najwięcej możliwości, a także możesz w największym stopniu wpłynąć na wydajność systemu SQL Server. Gdy wyłączysz komputer, jedyną istniejącą rzeczą są dane zapisane

na dysku twardym. Gdy włączysz zasilanie, procesor rozpocznie pracę, wczytany zostanie system operacyjny i uruchomi się system SQL Server. Wszystko to stanie się w wyniku wczytania danych i kodu z dysku.

Ten krótki opis dotyczy tego, co dzieje się na komputerze. Wszystko rozpoczyna się na dysku i musi zostać wczytane z niego do pamięci, a stamtąd do różnych buforów pamięci podręcznej procesora, skąd dane trafiają do procesora i mogą zostać użyte jako kod lub dane. Następnie wyniki uzyskane przez procesor muszą zostać ponownie zapisane na dysku, aby mogły przetrwać zdarzenia systemowe (np. zamknięcie, awarię lub konserwację systemu itd.).

System SQL Server bardziej niż wiele innych aplikacji jest wrażliwy na wydajność pracy dysku. Dzieje się tak, ponieważ zarządza dużymi ilościami danych przechowywanych w bazach użytkowników. Większość aplikacji może wczytać wszystkie dane z dysku w pamięci, a następnie działać przez długi czas bez konieczności ponownego dostępu do dysku. Twórcy systemu SQL Server też starają się umożliwić taki model działania, ponieważ jest to zdecydowanie najszybsze rozwiązanie. Niestety, gdy ilość danych potrzebna w określonej operacji przekracza pojemność pamięci, system SQL Server musi zacząć przenosić dane, aby uzyskać jak najwyższą wydajność. Rozpoczyna wtedy opróżnianie bufora zapisu i musi zapisywać dane z powrotem na dysk, aby zwolnić pamięć na generowane nowe wyniki.

W cyklu życia danych systemu SQL Server każda porcja danych musi wcześniej lub później zostać wczytana z dysku, a zmodyfikowane informacje trzeba z powrotem na nim zapisać.

Sieć

Wróć teraz do rysunku 10.8. Widać na nim, że sieć jest ważnym elementem systemu SQL Server. Sieć zapewnia połączenie, przez które system SQL Server odbiera wszystkie żądania i odsyła wyniki z powrotem do klientów. Współczesne szybkie sieci mają zwykle wystarczającą przepustowość, aby nie powodować zatorów i umożliwić systemowi SQL Server maksymalne wykorzystanie wszystkich pozostałych zasobów (procesora, pamięci i dysku).

Jednak w niektórych scenariuszach system SQL Server wykonuje stosunkowo niewiele pracy w porównaniu z liczbą żądań przesyłanych do serwera lub ilością danych zwracanych do klientów. W takich sytuacjach sieć może być wąskim gardłem. Zatory mogą powstawać w dowolnym miejscu sieci, np. w karcie sieciowej klienta, gdy klientem jest serwer aplikacji przekazujący do serwera bazodanowego setki tysięcy żądań na sekundę, lub w infrastrukturze sieci łączącej serwer bazodanowy z klientem (serwerem aplikacji, serwerem WWW lub stacją roboczą użytkownika). Infrastruktura sieci obejmuje liczne komponenty i może tworzyć proste systemy, w których dwa komputery są połączone w sieci LAN, a także bardzo skomplikowane połączone siecią systemy pracujące w internecie lub globalnych korporacyjnych sieciach WAN.

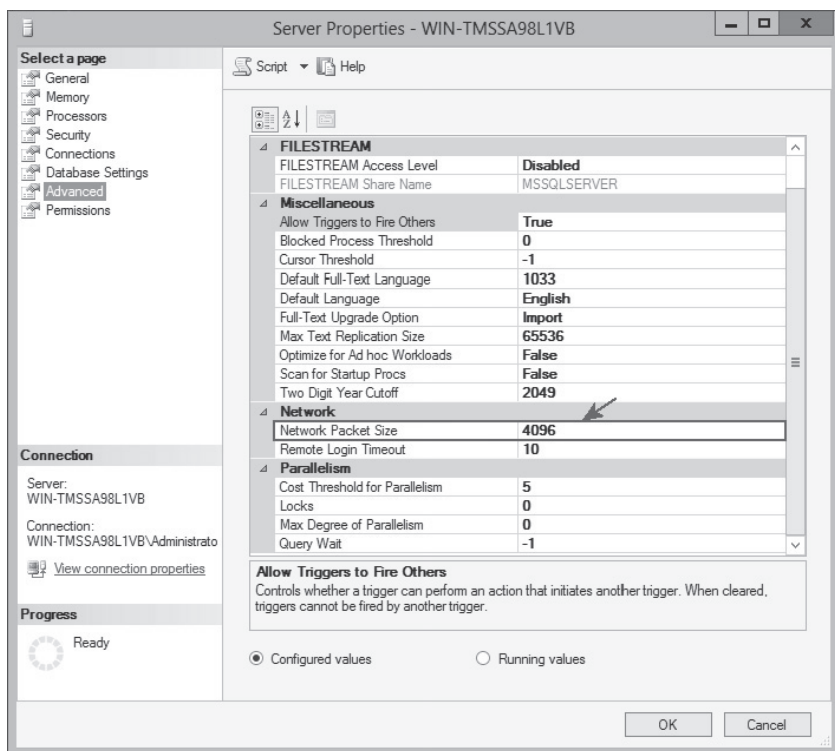
W większych, skomplikowanych systemach duża część sieci może pozostawać poza Twoją kontrolą i powodować nieakceptowalne problemy z przepustowością lub opóźnieniami. W takiej sytuacji możesz tylko zbadać sytuację, udokumentować ją i zgłosić.

W tym miejscu omawiamy elementy sieci, które możesz bezpośrednio kontrolować. Wszystkie one działają w ramach systemu SQL Server. Można założyć, że pozostałe części infrastruktury sieci potrafią obsłużyć zgłoszone żądania i odpowiednio szybko zwrócić wyniki do klienta.

Jedną z rzeczy, które należy sprawdzić, są ustawienia związane z szybkością i transmisją dwukierunkową. Niestety, łatwo niewłaściwie ustawić transmisję dwukierunkową i spowolnić pracę sieci. Standardowo sieć działa w trybie w pełni dwukierunkowym. Oznacza to, że dane są jednocześnie przesyłane w obu kierunkach. Wymaga to zastosowania odpowiednich kabli. Obecnie sieci działają z szybkością 10 GB/s.

W trakcie analizowania statystyk oczekiwania sieci często można natrafić na stan `ASYNC_NETWORK_IO`. Ten typ oczekiwania występuje, gdy system SQL Server zakończy przetwarzanie kwerendy nadesłanej przez aplikację użytkownika. Aby nie przeciążyć aplikacji klienta, system SQL Server wysyła początkowy pakiet i oczekuje, że aplikacja odbierze ten pakiet i odeśle potwierdzenie. W tym czasie stan oczekiwania to właśnie `ASYNC_NETWORK_IO`.

Jedną z możliwości zapobiegania temu stanowi jest upewnienie się, że pakiet o danej wielkości można przesłać za pomocą używanej infrastruktury sieci, i zmiana wielkości pakietu sieciowego na 16383 (jest to bardzo często używany rozmiar pakietów). Ilustruje to rysunek 10.9.



Rysunek 10.9. Zmiananie wielkości pakietu sieciowego

Dyski magnetyczne

W serwerach nadal najczęściej używane są dyski magnetyczne, dlatego warto je omówić. Są one jednak mniej wydajne od dysków SSD i Flash oraz macierzy takich dysków oferowanych przez Violin Memory i innych producentów. Dyski SSD, gdy są poprawnie skonfigurowane, działają szybciej niż magnetyczne dyski twarde. Odporność na błędy w stosowanych obecnie architekturach dysków SSD jest coraz lepsza, jeśli jednak przyjrzymy się cenom, dyski magnetyczne wciąż są bardziej ekonomicznym rozwiązaniem.

Innym ważnym obszarem wejścia-wyjścia są dyskowe operacje tego rodzaju. We wcześniejszych wersjach systemu SQL Server obsługa dysku była stosunkowo prosta, dlatego także liczba dostępnych opcji była niewielka. W większości sytuacji trzeba było zarządzać tylko kilkoma dyskami. Obecnie w systemach korporacyjnych można stosować technologie SAN (ang. *Storage Area Network*) i NAS (ang. *Network Attached Storage*), a także zewnętrzne podsystemy dyskowe

wykorzystujące macierze RAID (ang. *Redundant Array of Independent Disks*). Często używany jest też interfejs SCSI umożliwiający budowanie podsystemów obejmujących setki, a nawet tysiące dysków.

W systemach dyskowych używane są różne interfejsy.

- **ATA (ang. *Advanced Technology Attachment*)**. Inna nazwa to IDE (ang. *Integrated Drive Electronics*), która określa nie tylko działanie złącza i interfejsu, ale też to, że kontroler jest zintegrowany z napędem. Interfejsy PATA (ang. *Parallel ATA*) umożliwiają transfer danych równoległymi strumieniami między płytą główną a dyskiem z szybkością do 133 MB/s. Interfejs SATA (ang. *Serial ATA*) został opracowany w celu zlikwidowania architektonicznych ograniczeń interfejsu równoległego. SATA umożliwia transfer danych z szybkością do 3 GB/s.
- **SCSI (ang. *Small Computer Systems Interface*)**. Jest to zbiór standardów opracowanych dla połączeń i transferu danych między komputerami oraz licznymi urządzeniami peryferyjnymi, w tym dyskami. Interfejs wykorzystuje równoległe przesyłanie strumieni danych z szybkością do 640 MB/s. Rozwinięciem tego standardu jest SAS (ang. *Serial Attached SCSI*), w którym strumienie danych są przesyłane szeregowo z szybkością do 4800 MB/s.

Rozważmy teraz podstawowe czynniki fizyczne wpływające na wydajność dysków. Musisz poznać główne różnice między różnymi rodzajami dysków, ponieważ pozwala to zrozumieć różnice w ich wydajności. To z kolei pomoże wybrać odpowiedni typ dysków. W tabeli 10.3 przedstawione są przykładowe wartości (rejestrwane w idealnych warunków) typowych rodzajów opóźnienia dysków.

Tabela 10.3. Przykładowe opóźnienia dysków twardech

| Szybkość obrotowa dysku | Opóźnienie obrotowe | Czas przejścia do następnej ścieżki | Czas wyszukiwania | Szybkość przesyłu danych | Czas przesyłu 8 kilobajtów | Łączne opóźnienie |
|-------------------------|---------------------|-------------------------------------|-------------------|--------------------------|----------------------------|-------------------|
| 5400 RPM | 5,5 ms | 6,5 ms | 12 ms | 90 MB/s | 88 μs | 12,1 ms |
| 7200 RPM | 4,1 ms | 6,5 ms | 10,7 ms | 120 MB/s | 66 μs | 10,8 ms |
| 10 000 RPM | 3 ms | 1,5 ms | 4,5 ms | 166 MB/s | 48 μs | 4,6 ms |
| 15 000 RPM | 2 ms | 1,5 ms | 3,5 ms | 250 MB/s | 32 μs | 3,5 ms |

Większa liczba dysków

Większa liczba dysków pozwala uzyskać wyższą szybkość. Jeśli np. potrzebujesz 4 terabajtów pojemności, uzyskasz wyższą wydajność, gdy zastosujesz wiele małych dysków (10 dysków po 400 gigabajtów), niż gdy wykorzystasz dwa większe dyski (2 dyski po 2 terabajty). Wynika to z wielu przyczyn. Po pierwsze, mniejsze dyski są zwykle szybsze od większych. Po drugie, szybkość wzrasta, gdy do odczytu i zapisu danych używanych jest więcej wrzecion z wielu dysków pracujących w macierzy. Czasem pozwala to uzyskać przepustowość równą sumie przepustowości poszczególnych dysków. Jeśli wspomniane dyski o pojemności 400 gigabajtów i 2 gigabajtów działają z tą samą szybkością (np. 20 MB/s), dla 2 dysków suma wyniesie tylko 40 MB/s. Jednak dla 10 mniejszych dysków suma jest równa 200 MB/s, czyli pięciokrotnie więcej.

Szybsze dyski

Nie jest zaskoczeniem, że szybsze dyski zapewniają wyższą wydajność niż wolniejsze. Jednak nie chodzi tu wyłącznie o szybkość obrotową, która jest tylko jednym z czynników wpływających na ogólną prędkość dysku. Istotne jest to, czy dysk poradzi sobie z obsługą operacji wejścia-wyjścia w procesach pracy, które są najważniejsze dla użytkowników.

Niestety, producenci dysków rzadko (jeśli w ogóle) podają dane inne niż prędkość obrotowa i teoretyczna szybkość magistrali dysku. Przykładowo często można przeczytać, że dysk SCSI o szybkości 10 000 lub 15 000 RPM ma maksymalną przepustowość 300 MB/s, ponieważ używa magistrali SCSI 320. Jeśli jednak podłączysz taki dysk i przeprowadzisz testy przy użyciu programu SQLIO lub podobnego narzędzia, bardzo możliwe jest, że dla odczytu losowych niewielkich bloków nieprzechowywanych w kolejce uzyskasz przepustowość na poziomie 2 – 4 MB/s. Nawet dla najszybszych operacji wejścia-wyjścia, czyli sekwencyjnych operacji na dużych blokach, przepustowość rzadko będzie przekraczać 60 – 70 MB/s. To dużo mniej niż 300 MB/s.

UWAGA Narzędzie SQLIO Disk Subsystem Benchmark możesz pobrać ze strony www.microsoft.com/download/en/details.aspx?id=20163. Pod adresem http://sqlserverpedia.com/wiki/SAN_Performance_Tuning_with_SQLIO znajdziesz bardzo dobry samouczek, za pomocą którego zapoznasz się z tym programem.

Odczyt i zapis w pamięci podręcznej

W każdym dysku twardym znajduje się obszar pamięci nazywany *buforem dysku* lub *pamięcią podręczną dysku*. Pamięć podręczna pełni funkcję bufora między dyskiem a podłączonym systemem i służy do przechowywania danych wczytywanych z dysku lub zapisywanych na nim. Pojemność tego typu pamięci podręcznej wynosi od 8 do 64 megabajtów. Także kontrolery dysków (niezależnie od tego, czy działają wewnętrznie na serwerze, czy zewnętrznie w ramach architektury SAN) mają pamięć podręczną używaną do przechowywania wczytywanych i zapisywanych danych. Pojemność pamięci podręcznej kontrolera wynosi zwykle od 512 megabajtów do 512 gigabajtów.

Oto wybrane z wielu zastosowań pamięci podręcznej dysku.

- **Odczyt z wyprzedzeniem i odczyt z opóźnieniem.** Gdy dysk ma wczytać dane, może pobrać także informacje, których nie zażądano, jeśli uzna, że mogą być później przydatne systemowi SQL Server.
- **Przyspieszanie zapisu.** Kontroler dysku może natychmiast po otrzymaniu danych poinformować system SQL Server, że operacja zapisu się powiodła, choć dane nie są jeszcze zapisane na dysku. Dane trafiają wtedy do pamięci podręcznej zapisu i są zapisywane później. Grozi to utratą danych, jeśli nastąpi przerwa w zasilaniu przed fizycznym zapisaniem danych. Aby ograniczyć to zagrożenie, większość kontrolerów macierzy dyskowych ma bateryjne zasilanie awaryjne, co gwarantuje zakończenie wszystkich zaległych operacji zapisu nawet wtedy, gdy główne zasilanie zostanie wyłączone.
- **Równoważenie operacji wejścia i wyjścia.** Częstotliwość wykonywania operacji odczytu i zapisu waha się w trakcie pracy systemu. Aby mieć pewność, że żądania zostaną odpowiednio szybko obsłużone, używa się pamięci podręcznej do przechowywania danych oczekujących na przesłanie w obu kierunkach.

W większości konfiguracji pamięć podręczna odczytu i zapisu korzysta z tego samego obszaru pamięci, a procesor dysku (lub macierzy) kontroluje szczegóły jej działania. W niektórych konfiguracjach baz OLTP pamięć podręczna odczytu jest wyłączona, co pozwala zwiększyć ilość pamięci podręcznej zapisu.

Dyski SSD (dyski z pamięcią flash)

Trudno porównywać dyski SSD z dyskami magnetycznymi. Tradycyjne pomiary dysków twardej dotyczą głównie cech, w których takie dyski nie osiągają dobrych wyników (np. opóźnienia obrotowego lub czasu wyszukiwania). Dyski SSD nie muszą się obracać, dlatego nie mają opóźnienia obrotowego. W takich dyskach mogą jednak występować problemy z naprzemiennymi operacjami odczytu i zapisu. Ponadto z czasem wydajność takich dysków spada.

Istnieją dwa rodzaje pamięci flash — NAND i NOR. Pamięć NOR działa podobnie jak pamięć RAM i wymaga zasilania, aby przechowywać dane. Pamięć NAND jest trwała i nie wymaga elektryczności do przechowywania danych. Dlatego skoncentrujemy się tu na pamięci NAND.

Dyski SSD typu NAND obejmują chipy z pamięcią flash dotyczącą komórek SLC (ang. *Single Layer Cells*) lub MLC (ang. *Multi Layer Cells*). Komórki SLC przechowują tylko jeden bit danych (0 lub 1). Komórki MLC mogą przechowywać więcej informacji. Obecnie zwykle zawierają po 2 bity. Gęstość danych w komórkach MLC jest wyższa, dlatego dyski z takimi komórkami są bardziej pojemne od dysków z komórkami SLC.

Odczyt danych z dysków z komórkami MLC może być wolniejszy z powodu korekcji błędów niezbędnej w celu naprawiania pomyłek. Takie dyski wymagają zapisu całych stron, natomiast w pamięciach SLC możliwy jest wydajniejszy zapis części stron. Jednak większość firm odchodzi od dysków SLC SSD w kierunku dysków MLC SSD. Strona na dyskach SSD ma zwykle 4 kilobajty. Strony są zapisywane w blokach mających przeważnie od 128 do 512 kilobajtów.

Początkowy zapis na dysku SSD jest mniej kosztowny, ponieważ zapisywana jest tylko jedna strona. Jednak późniejsze operacje zapisu, np. przy usuwaniu i ponownym dodawaniu danych, wymagają opróżnienia strony i ponownego zapisania bloku. Jeśli chodzi o wydajność, odczyt zajmuje zwykle 25 mikrosekund, zapis pierwszej strony o wielkości 4 kilobajtów trwa 250 mikrosekund, a usunięcie i ponowne zapisanie bloku o pojemności 256 kilobajtów wymaga 2 milisekund.

Dyski SSD po pewnym czasie się zużywają. Można na nich wykonać tylko określoną liczbę operacji. Dyski SLC SSD zwykle zaczynają się psuć po 100 000 cykli kasowania, a dyski MLC SSD zaczynają sprawiać problemy po 5000 – 10 000 takich cykli. Jeśli system co sekundę aktualizuje i kasuje jeden blok, wykonanie 10 000 cykli zajmie ponad 5 lat. Jeśli jednak te operacje dotyczą zawsze tego samego bloku, zużycie nastąpi już po 3 godzinach.

Zaletą dysków SSD jest to, że gdy zapis staje się niemożliwy, nadal można odczytać dane.

Rozważania związane z przechowywaniem danych

Po omówieniu różnych komponentów systemu przechowywania danych pora zastanowić się, jak najlepiej go skonfigurować. Jest to trudne zadanie, ponieważ nie istnieje jeden prosty sposób na skonfigurowanie tego systemu, sprawdzający się w każdej sytuacji. System SQL Server jest używany do wykonywania bardzo zróżnicowanych zadań, dlatego każdy jego egzemplarz może wymagać zupełnie innej konfiguracji pamięci, dostosowanej do konkretnych wymagań z zakresu

wejścia-wyjścia. W kolejnych podpunktach przedstawiamy zbiór wskazówek, a następnie szczegółowo opisujemy, jak wybrać najlepsze rozwiązanie. Wskazówki możesz wykorzystać w trakcie dyskusji z zespołem odpowiedzialnym za przechowywanie danych.

Wykorzystaj wiedzę producentów

Osobami, do których powinieneś się zwrócić o poradę w obszarze konfigurowania różnych komponentów, są ich producenci. Możliwe jednak, że oni nie będą wiedzieć, jak skonfigurować sprzęt pod kątem systemu SQL Server. Dlatego musisz przekazać im wymagania związane z serwerem SQL Server w zrozumiały sposób. Najlepiej podać w tym celu stosunek liczby operacji zapisu do odczytu i operacji wejścia-wyjścia z dostępem bezpośrednim do operacji sekwencyjnych, wielkość bloków, liczbę operacji wejścia-wyjścia na sekundę, oczekiwaną przepustowość w megabajtach na sekundę, a także minimalne i maksymalne opóźnienie. Te informacje pomogą dostawcy w określeniu optymalnych ustawień dla oferowanych komponentów: dysków, kontrolerów macierzy, kabli, sprzętu sieciowego i innych elementów związanych z przechowywaniem danych.

Każdy system jest inny

Każdy egzemplarz systemu SQL Server ma inne wymagania dotyczące operacji wejścia-wyjścia. Pamiętaj o tym i nie próbuj stosować gotowych schematów w konfiguracji systemu wejścia-wyjścia (chyba że przeprowadziłeś wcześniej analizy i stwierdziłeś, iż Twój egzemplarz i egzemplarz wzorcowy mają dokładnie takie same wymagania).

Im prościej, tym lepiej

W inżynierii od dawna uważa się, że proste rozwiązania są łatwiejsze do zaprojektowania, zbudowania, zrozumienia i konserwacji. W większości przypadków dotyczy to także projektów systemu wejścia-wyjścia. Prostsze rozwiązania działają szybciej, są bardziej niezawodne i wymagają mniej konserwacji niż bardziej skomplikowane projekty. Jeśli nie masz konkretnych powodów do tworzenia skomplikowanego projektu systemu przechowywania danych, zachowaj prostotę.

Testy

Testy są nieodzownym elementem konfigurowania, optymalizowania i dostrajania wydajności. Zbyt często można zetknąć się z klientem, który wyłącznie na podstawie własnych przeczuć lub zasłyszanych gdzieś półprawd jest przekonany, że białe jest czarne.

Jeśli nie masz wyników testów, naprawdę nie wiesz, jak działa podsystem wejścia-wyjścia. Zapomnij o spekulacjach i zgadywaniu. Zaczynaj testować system wejścia-wyjścia, aby ustalić, co się w nim dzieje. Często używanym narzędziem do pomiarów podsystemu wejścia-wyjścia i badania charakterystyki jest IOMeter. Generuje on obciążenie symulujące operacje wejścia-wyjścia na dysku lub w sieci i wykonuje pomiary. Można je pobrać ze strony <http://www.iometer.org>.

UWAGA Dobry samouczek pomocny przy poznawaniu narzędzia IOMeter znajdziesz na stronie <http://www.techrepublic.com/article/test-storage-system-performance-with-iometer/5735721>.

Często stosowaną miarą wydajności dysku jest opóźnienie. Można je zmierzyć w monitorze wydajności systemu Windows za pomocą liczników *Średni czas dysku w s/Odczyt*, *Średni czas dysku w s/Zapis* i *Średni czas dysku w s/Transfer*. Oto docelowe wartości opóźnienia dysku.

- **Dziennik transakcji bazy danych.** Mniej niż 5 milisekund (najlepiej 0 milisekund).
- **Dane OLTP.** Mniej niż 10 milisekund.
- **Dane w systemach wspomagających podejmowanie decyzji (OLAP i generowanie raportów).** Mniej niż 25 milisekund.

Po przygotowaniu i przetestowaniu systemu trzeba go monitorować, aby wykryć zmiany w działaniu i wydajności podsystemu wejścia-wyjścia bezpośrednio po ich wystąpieniu. Jeśli masz politykę monitorowania, możesz utworzyć historię wydajności systemu. Będzie ona nieoceniona przy późniejszych analizach wydajności. Bez solidnej historii uzyskanej za pomocą monitorowania danych trudno prześledzić początki wolno zmieniającego się trendu. Szczegółowe informacje o tym, co i jak warto monitorować, znajdziesz w rozdziale 13., „Dostrajanie wydajności kodu w języku T-SQL”.

Macierze RAID

W trakcie zastanawiania się nad tym, ile dysków potrzebujesz, pomyśl o poziomie RAID, który chcesz zastosować. Może on wpłynąć na liczbę dysków niezbędnych do zbudowania systemu przechowywania danych o określonej pojemności i cechach z zakresu operacji wejścia-wyjścia. Rozważ następujące kwestie.

- **Dostępność.** Pierwszym czynnikiem analizowanym w kontekście macierzy RAID jest poziom dostępności danych.
- **Cena.** Ważną kwestią przy tworzeniu każdego systemu jest zmieszczenie się w budżecie. Nie ma sensu tworzyć specyfikacji obejmującej najnowszy, najbardziej wydajny system, jeśli jego cena przekracza budżet 10, 100, a nawet 1000 razy.
- **Przestrzeń fizyczna.** Innym ważnym czynnikiem połączonym z kosztami jest to, ile przestrzeni fizycznej będziesz potrzebował.
- **Wydajność.** Wydajność systemu przechowywania danych to następny istotny aspekt, który pomaga ustalić docelowy poziom RAID.

RAID 0 — przeplatanie bez bloków parzystości i kopii lustrzanych

Macierz RAID 0 obejmuje przynajmniej dwa dyski. W tym podejściu dane są przeplatane między wszystkimi dyskami. Ten poziom RAID nie zapewnia nadmiarowości ani odporności na błędy, ponieważ awaria jednego dysku uszkadza macierz. W trakcie zapisu dane są dzielone na bloki, które są równolegle umieszczane na dyskach.

Pozwala to zwiększyć szybkość odczytu, ponieważ można równolegle wczytywać wiele fragmentów określonych danych. Jednak RAID 0 nie obejmuje kontroli błędów. Ten poziom nie jest zalecany dla systemów SQL Server.

RAID 1 — kopie lustrzane z przeplotem lub blokami parzystości (dwa dyski)

W macierzach RAID 1 jeden dysk jest kopiowany na drugi. To oznacza, że takie macierze muszą obejmować dwa dyski. Ten model zapewnia szybkie działanie, ponieważ dane można wczytywać (choć nie zawsze) z obu dysków, a spadek wydajności przy zapisie jest minimalny. RAID 1 zapewnia nadmiarowość i odporność na błędy, jednak zwiększa koszty przechowywania danych, ponieważ można wykorzystać tylko 50% łącznej pojemności dysków. Z powodu kosztów można stwierdzić, że operacje tworzenia kopii zapasowych i wczytywania danych nie wymagają ochrony na tym poziomie.

RAID 10 — przeplot z kopiami lustrzanymi (przynajmniej cztery dyski)

RAID 10 (inna nazwa to RAID 1+0) polega na zastosowaniu przeplotu dla kopii lustrzanych dysków, co wymaga przynajmniej czterech dysków. Macierz tego typu zawsze składa się z parzystej liczby dysków.

Ogólnie jest to najszybsze rozwiązanie. Macierze RAID 5 (opisane poniżej) zapewniają jednak szybszy odczyt, jeśli obejmują tyle samo dysków. W RAID 10 dane są wczytywane z kilku dysków, a zapis prowadzi do tylko minimalnego spadku wydajności. Ten poziom zapewnia też nadmiarowość — macierz może przetrwać awarię więcej niż jednego dysku, przy czym uszkodzone nie mogą być dyski zawierające te same dane. Wadą jest wyższy koszt, ponieważ można wykorzystać tylko 50% całej dostępnej pamięci.

Systemy bazodanowe wymagające wysokiej wydajności odczytu i zapisu oraz nadmiarowości należy przechowywać w macierzach RAID 10. Z powodu kosztów operacje tworzenia kopii zapasowych, wczytywania danych i inne operacje tylko do odczytu mogą nie wymagać ochrony na tym poziomie. Alternatywą dla macierzy RAID 1+0 jest macierz RAID 0+1, w której tworzone są kopie lustrzane dla dysków z przeplotem (w odróżnieniu od poziomu RAID 1+0, gdzie tworzony jest przeplot dla kopii lustrzanych).

RAID 5 — przeplot z blokiem parzystości (minimum trzy dyski)

W macierzach RAID 5 stosowany jest przeplot z blokiem parzystości i niezbędne są przynajmniej trzy dyski. W trakcie zapisu obliczana jest parzystość danych. Przykładowo dla każdej operacji zapisu w trzydyskowej macierzy dane są umieszczane na dwóch dyskach, a na trzecim dysku zapisywana jest parzystość. Oprogramowanie macierzy RAID rozdziela bloki parzystości między wszystkie dyski, aby uniknąć przeciążenia operacjami zapisu.

Obliczanie parzystości prowadzi do spadku wydajności, dlatego macierze RAID 5 nie są dobrym rozwiązaniem do przechowywania baz danych, które muszą obsługiwać wiele operacji zapisu. Inną wadą macierzy tego typu jest to, że awaria dysku może prowadzić do znacznego spadku wydajności w czasie odtwarzania danych na nowym dysku. W trakcie pracy macierzy z uszkodzonym dyskiem wydajność jest niższa także z powodu konieczności obliczania parzystości przy każdym odczycie danych.

Przy odczycie macierze RAID 5 mogą działać szybciej niż inne macierze RAID, ponieważ grupa dysków może równolegle udostępniać dane. Dlatego RAID 5 to dobry wybór dla baz danych używanych głównie do odczytu (np. w systemach wspomagania podejmowania decyzji). Ponadto macierze RAID 5 są bardziej ekonomiczne od macierzy RAID 1 i RAID 10, ponieważ na każdą macierz wymagają tylko jednego dysku do przechowywania bloków parzystości. W macierzach RAID 1 i RAID 10 nadmiarowe dane zajmują aż 50% przestrzeni.

RAID 6 — przeplot z podwójną parzystością (minimum cztery dyski)

Macierz RAID 6 obejmuje przynajmniej cztery dyski, na których przechowywane są dwie kopie bloków parzystości. Zapewnia to wyższą odporność na błędy, ponieważ awaria nawet dwóch dysków nie musi prowadzić do uszkodzenia danych w macierzy. W tym modelu praktyczne jest tworzenie macierzy obejmujących wiele dysków, ponieważ dyski o dużej pojemności wydłużają czas przywracania systemu po awarii.

Jednak nie należy korzystać z macierzy RAID 6. Wykorzystywana jest w nich macierz z podwójną parzystością. W macierzy RAID 5 używany jest pojedynczy blok parzystości, natomiast tu używane są dwa takie bloki. Choć pozwala to przywrócić dane po awarii dwóch dysków, powoduje też obniżenie wydajności.

Rekomendowane poziomy RAID

Dla plików danych i plików dzienników systemu SQL Server powinieneś stosować szybki i niezawodny system przechowywania. W większości egzemplarzy systemu SQL Server dla takich plików zaleca się stosowanie przeplotu z kopiami lustrzanymi (RAID 10).

Gdy nie można zastosować macierzy RAID 10, należy zbudować macierz RAID 5. Jeśli dana baza jest używana w aplikacji, która wykonuje dużo operacji odczytu w porównaniu z liczbą operacji zapisu, lub gdy baza działa w trybie tylko do odczytu, macierz RAID 5 z dużą liczbą dysków to akceptowalne rozwiązanie.

Ponadto jeśli wiesz, że aplikacja będzie często korzystać z bazy tempdb, zastosuj możliwie szybką i niezawodną pamięć. Może się to wydawać dziwne, ponieważ dane w bazie tempdb są tymczasowe, jednak wymóg niezawodności związany jest z zapewnianiem ciągłości pracy systemu, a nie z obawą o utratę danych. Jeśli dla reszty systemu używana jest niezawodna pamięć, a baza tempdb zapisana jest w innym miejscu, awaria jednego dysku może uniemożliwić pracę systemu SQL Server. Wymóg wysokiej szybkości wynika z tego, że na bazie tempdb wykonywanych jest wiele operacji wejścia-wyjścia z dostępem bezpośrednim.

Pliki binarne systemu operacyjnego i systemu SQL Server mogą być zapisane w zwykłej kopii lustrzanej, choć w wielu sytuacjach czas przestoju, gdy konieczne jest przywrócenie tych systemów, jest akceptowalny. Wtedy wystarczy wykorzystać jeden dysk.

OSTRZEŻENIE Pamiętaj: jeśli nawet czas potrzebny na przywrócenie i zainstalowanie poprawek systemu operacyjnego, zainstalowanie systemu SQL Server i poprawek do niego oraz podłączenie baz danych jest krótszy niż limit czasu przestoju, koszt zatrudnienia pracowników niezbędnych do wykonania wszystkich tych zadań jest znacznie wyższy od kosztu drugiego dysku w macierzy RAID 1.

W systemach o znaczeniu strategicznym pliki binarne systemu operacyjnego i systemu SQL Server powinny znajdować się w kopii lustrzanej w macierzy, przy czym wystarczy jedna taka kopia. Dla plików tego rodzaju wymogi związane z operacjami wejścia-wyjścia nie są wysokie. Pliki te są zwykle wczytywane raz w momencie uruchamiania aplikacji. Potem nie trzeba ich modyfikować do momentu, gdy potrzebne będą nowe ścieżki kodu. Wtedy następuje odczyt bezpośredni kilku kolejnych porcji danych po 4 kilobajty, dlatego stosowanie szybkich dysków nie jest tu konieczne.

Podsumowanie

Dostrajanie wydajności bywa skomplikowane, gdy system przez dłuższy czas działa poniżej możliwości. Problemy z wydajnością narastają, jeśli system od początku nie został zaprojektowany w optymalny sposób.

Zanim zaczniesz modyfikować sprzęt, powinieneś zastanowić się nad wieloma obszarami, takimi jak interakcja użytkowników z systemem, wzorce użytkowania danych, liczba i rodzaj instrukcji SQL-a uruchamianych w systemie, schemat itd.

W trakcie konfigurowania serwera pod kątem optymalnej wydajności trzeba podjąć szereg decyzji dotyczących sprzętu. Procesor serwera ma duży wpływ na to, czy system będzie akceptowalnie działał przy zakładanym obciążeniu. Należy zastanowić się nad komponentami

procesora (pamięcią podręczną, technologią Hyper-Threading, wielordzeniowością i architekturą), a następnie rozważyć dostępne opcje. Pomyśl też o pamięci i różnych związanych z nią mechanizmach. Te mechanizmy to m.in. fizyczna i wirtualna przestrzeń adresowa, menedżer pamięci wirtualnej i plik stronicowania.

Najwolniejszą częścią systemu jest podsystem wejścia-wyjścia. Należy starannie wybrać jedną z wielu możliwości dostępnych przy projektowaniu podsystemów sieci i przechowywania danych. Oto niektóre z pytań, na które trzeba udzielić odpowiedzi. Jak szybka powinna być sieć? Ile przestrzeni dyskowej jest potrzebne? Ile dysków i jakiego rodzaju należy użyć? Jak szybkie mają być te dyski? Czy należy zastosować architekturę SAN, czy NAS? Jakiego typu macierz RAID wykorzystać? Jakich adapterów pamięci użyć? Jaka powinna być wielkość jednostek alokacji pamięci?

Znasz już wiele narzędzi, które możesz wykorzystać w trakcie konfigurowania systemu SQL Server w celu uzyskania optymalnej wydajności.

Z rozdziału 11. dowiesz się, jak optymalizować system SQL Server.

Skorowidz

.NET, 221
.NET CLR Exceptions, 229
.NET CLR Loading, 229
.NET CLR Memory, 229

A

Activity Monitor, 132
administrator
 baz danych, 37
 BI, 38
 produkcyjnej bazy danych, 291
 programista, 289
administrowanie bazami SQL Database, 795, 809
adres URL
 narzędzia Report Manager, 741
 URL usługi sieciowej, 737
agenty replikacji, 475
akcje, 385
aktualizacja
 aktywnego serwera, 672, 673
 katalogu pełnotekstowego, 109
 klastrow, 524, 527
 systemu, 91, 94, 105
 typu Cumulative Update, 544
 usług Reporting Services, 109
aktualizowanie, 67
 do wersji 64-bitowej, 110
 przesyłania dzienników, 639
 w miejscu, 94, 97
aktywna replika pomocnicza, 839
aktywność dysków, 360
alerty, 166, 317, 668
 dotyczące wydajności, 169
 dotyczące zdarzeń, 170
aliasy dla klientów, 634
alokacja plików, 318
AlwaysOn, 817, 844
analizowanie
 metadanych, 843
 operacji, 470
 wydajności kwerend, 444
analizy
 biznesowe, 38
 następcze, 731
aplikacja Secure Store Service, 778
aplikacje POS, 480
aplikacje warstwy danych, 563
APS, Analytic Platform System, 29
architektura
 bazy SQL Database, 796
 grup dostępności, 818
 kopii lustrzanych, 644
 mechanizmu SQL Trace, 375, 376
 NUMA, 296, 334
 P2P, 481, 482
 przesyłania dzienników, 611
 serwera SSAS, 714
 systemu, 35, 39, 772
 WSSRA, 288
 x64, 294
archiwizowanie danych, 606
ARM, Analyze, Report, Migrate, 249
asynchroniczne
 aktualizowanie statystyk, 453
 przesyłanie komunikatów, 189
ATA, Advanced Technology Attachment, 303
automatyczne przełączanie awaryjne, 657

automatyzowanie
 pracy, 147
 systemu, 156
 autoryzacja dostępu, 237

B

baza danych
 rozmieszczanie plików, 313
 baza danych
 master, 42, 576
 MDW, 267, 292
 model, 43, 84, 577
 msdb, 43, 576
 resource, 41
 tempdb, 42, 83, 314, 577
 SQL Database, 796, 803
 SSRS, 739
 bazy danych
 częściowo niezależne, 120
 systemowe, 122
 BCP, Bulk Copy Program, 557
 bezpieczeństwo, 231
 BI, Business Intelligence, 36, 679
 BISM, BI Semantic Model, 794
 blokada
 indeksu, 453
 partycji tabeli, 321
 błąd strony, 298, 300
 błąd strony miękkiej, 300
 BPE, Buffer Pool Extension, 339
 bufor, 339

C

cele monitorowania, 354
 centralne serwery zarządzania, 139
 chmura, 36, 795
 CLR, Common Language Runtime, 219
 CTE, Common Table Expression, 83
 czas trwania przełączania awaryjnego, 661
 częstotliwość odświeżania danych, 791

D

DAC, Data Tier Application, 563
 dane
 replikacji, 475
 uwierzytelniające, 174, 789
 XML, 453
 zewnętrzne, 779

DAS, Direct-Attached Storage, 61
 definicja wydajności, 287
 definiowanie obciążenia, 312
 DMO, Dynamic Management Objects, 44, 113
 DMV, 231
 dodatek PowerPivot, 783
 dodawanie
 baz danych, 834
 replik, 831
 replik w platformie Azure, 832
 systemu do klastra, 537
 środowiska dla projektu, 699
 węzła, 503
 wiersza, 254
 domeny aplikacji, 221
 domyślny ślad, 357, 371, 373
 dostawcy danych, 778
 dostęp
 do danych, 427
 do pamięci, 343
 do replik pomocniczych, 821
 do serwera, 807
 do zewnętrznych danych, 778, 789
 tylko do odczytu, 839
 dostępność, 583
 dostrajanie
 pamięci, 341
 wydajności, 280
 wydajności kodu, 415
 wydajności kwerend, 420
 drzewo parsowania, 417
 DSS, Decision Support System, 289
 DTA, Database Tuning Advisor, 320, 454, 468
 DW, Data Warehousing, 289
 dynamicznie dodawane procesory, 334
 dysk, 62, 304, 360, 362
 magnetyczny, 302
 opóźnienia, 306
 pamięć podręczna, 304
 RAID, 307
 SSD, 62, 305
 dystrybutor, 474, 483
 działanie
 indeksów, 451–454
 klastrowania, 517
 dzielenie partycji, 329
 dziennik, 211, 410, 607
 błędów, 131, 410, 746
 transakcji, 39, 571
 wykonywania raportów, 746
 zdarzeń, 411, 687

E

edycje systemu
 Business Intelligence Edition, 53
 Enterprise Edition, 52
 Standard Edition, 53
 egzemplarz systemu, 520
 ekran
 Database Engine Configuration, 69
 New Session, 397, 398, 399
 Server Properties, 126
 eksportowanie pakietów, 690
 ekstenty, 318
 dedykowane, 318
 niejednorodne, 318
 Excel
 dodatek PowerPivot, 770, 783

F

filtr, 386
 filtrowanie
 obiektów, 131
 publikowanych danych, 489
 filtry sesji zdarzeń, 394
 fizyczna przestrzeń adresowa, 298
 formaty planów, 423
 fragmentacja indeksu, 462
 funkcja
 AlwaysOn Availability Groups, 36
 DMF, 400
 PowerPivot for SharePoint, 80
 Resource Governor, 261
 suser_sname(), 247
 sys.dm_exec_sql_text, 138
 funkcje, 38, 243
 DMF, 357
 kopii lustrzanych, 659
 nieobsługiwane, 106
 partycjonujące, 322, 323
 przestarzałe, 108
 przywracania, 600
 replik pomocniczych, 839

G

GAM, Global Allocation Map, 318
 generowanie
 klucza, 786
 planów wykonywania, 418
 planów wykonywania kwerend, 423

raportów, 610
 snapshota, 490
 tabel, 812
 widoków, 812
 gniazdo, 296
 grupa
 plików, 324
 konwersacji, 202
 operacyjna, 346
 serwerów, 139
 grupy dostępności, 241, 676, 817
 AlwaysOn, 516
 architektura, 818
 dodawanie baz, 834
 konfigurowanie, 823
 monitorowanie, 846
 panel kontrolny, 844, 845
 przełączanie awaryjne, 834
 repliki, 819
 role, 819
 rozwiązywanie problemów, 846
 sprawdzanie konfiguracji, 829
 ustawianie baz, 825, 834
 włączanie obsługi, 823

H

harmonogram, 162
 odświeżania pamięci podręcznej, 766
 przetwarzania subskrypcji, 764
 historia
 kopii zapasowych, 574
 raportu, 766

I

IAM, Index Allocation Map, 250, 318
 IDE, Integrated Drive Electronics, 303
 identyfikowanie przyczyn zakleszczenia, 372
 ilość pamięci, 343
 implementowanie replikacji, 483
 migawkowej, 487
 transakcyjnej, 500
 importowanie
 danych, 789
 pakietów, 690
 indeksowanie baz danych, 449
 indeksy, 253
 filtrowane, 451, 456
 klastrowane, 429, 455

indeksy

- nieklastrowane, 432, 453, 455
- nieużywane, 406
- oparte na kolumnach, 37, 450, 456
- oparte na przedziałach, 256
- pełnotekstowe, 453, 458, 577
- podzielone na partycje, 442, 452, 454, 461
- pokrywające, 455
- przechowywane w pamięci, 458
- przestrzenne, 452, 459
- wierszowe, 455
- XML-owe, 451, 458
- z haszowaniem, 253

informacje

- o blokadach, 404
- o brakujących indeksach, 468
- o kopiach lustrzanych, 666
- o podłączonych użytkownikach, 408
- o połączeniu, 760
- o poziomie fragmentacji indeksów, 463
- o procedurze składowanej, 278
- o skanowaniu tabeli, 431
- o stanie subskrypcji, 507
- o tabeli, 276
- o zerowaniu plików, 286

inicjowanie plików

- plików bazy danych, 83, 284
- snapshota, 498

In-Memory OLTP, 249

- indeksy, 253
- rekordy, 252
- struktury tabel, 252
- wirtualizacja, 258

inspekcja działań użytkowników, 372

instalacje

- nienadzorowane, 70
- nadzorowane, 67
- równoległe, 66

instalowanie

- funkcji PowerPivot, 80
- klastra, 531, 532
- pakietów, 691, 696
 - eksportowanie, 694
 - importowanie, 694
 - narzędzie DTUtil, 694
 - przenoszenie, 694
 - zapisywanie, 694

- systemu, 66
- usług, 76
- węzłów klastra, 533

instrukcja

- cmpxchg16b, 258
- DELETE, 441
- DENY, 237
- GRANT, 237
- RESTORE, 603
- REVOKE, 237

instrukcje

- do testowania wydajności, 425
- instalowania systemu, 538
- SQL, 290
- tworzenia kopii zapasowych, 592
- wdrażania przesyłania dzienników, 624

integracja

- z CLR, 222, 231
- z SharePoint, 769

izolacja snapshotów, 841

J

jednostki

- docelowe, 387
- pośredniczące, 175

język

- ASSL, 717
- SQL, 290
- T-SQL, 38, 136, 157, 222, 415

K

kardynalność, 418, 419

katalog SSIS, 695

kategorie zadań, 158

klastr

- dodawanie systemu, 537
- grupy zasobów, 540
- instalacja systemu, 538
- instalowanie węzłów, 533
- monitorowanie, 546
- problemy, 547
- ustawianie dysków, 541
- używanie koordynatora MSDTC, 536
- Windows Server 2012 R2, 532, 535
- WSFC, 532, 533, 614, 635
- z wieloma egzemplarzami, 522, 524
- z dwoma węzłami, 522, 543

klastrowanie, 511

- przygotowywanie infrastruktury, 528
- przygotowywanie sprzętu, 528
- w Windows Server 2012 R2, 531

- klucz główny, 208, 786
- klucze szyfrowania, 744
- kolacja, collation, 58, 63
 - systemu SQL Server, 64
 - systemu Windows, 64
- kolejki, 197
- kolejki komunikatów, 192
- kolejność sortowania, 64
- koligacja, 336
- komórki
 - MLC, 305
 - SLC, 305
- komponenty
 - CLR, 223
 - przepływu danych, 682
 - systemu, 109
- kompresja, 37, 330
 - danych, 325, 331, 332
 - indeksów, 451
 - kopii zapasowej, 569
 - przedrostkowa, 328
 - słownikowa, 328
 - stron, 328
 - tabel, 451
 - wierszy, 326
- komunikat o błędzie, 808
- konfigurowanie
 - alertu, 670
 - baz MDW, 267
 - baz SQL Database, 798
 - dystrybucji, 483
 - grupy dostępności, 823
 - harmonogramu przetwarzania subskrypcji, 764
 - inicjowania snapshota, 498
 - istniejących grup dostępności, 830
 - katalogu SSIS, 695, 696
 - konta usługowego, 736
 - narzędzia SQL Server Agent, 176
 - narzędzia SSS, 786
 - nośników, 586
 - nowego alertu, 317
 - nowych subskrypcji, 757
 - obsługi błędów, 725
 - odbiornika, 828
 - odświeżania danych, 792
 - opcji serwera, 542
 - pakietów, 697
 - pakietu SSIS, 568
 - parametrów pakietu, 700
 - plików przywracanej bazy, 601
 - po instalacji, 81
 - publikacji snapshota, 488
 - replikacji migawkowej, 483
 - replikacji w trybie P2P, 501
 - Service Brokera, 193
 - serwera, 279
 - serwera lustrzanego, 670
 - SharePointa, 771
 - sieci, 541
 - subskrypcji, 493, 759
 - synchronizacji danych, 829
 - systemu, 70, 126, 282
 - tworzenia kopii zapasowych, 843
 - udziału plikowego, 762
 - usługi SSIS, 684
 - ustawień systemu, 85
 - właściwości planu konserwacji, 149
 - zadania, 151, 155
 - zapory, 806
 - źródła danych, 751
- konserwacja, 582
 - bazy danych, 589
 - indeksów, 461
- konto
 - domenowe, 743
 - SA, 85
 - usługowe, 65, 171
 - wykonawcze, 743
- kontrakty, 196
- kontrola
 - wersji wierszy, 83
 - zmian, 683
- kontrolery dysków, 63
- kończenie przesyłania dzienników, 637
- koordynator MSDTC, 536
- kopia, 556
 - częściowa, 553
 - pełna, 553
 - zapasowa, 86
 - zapasowa bazy SSAS, 727
- kopie lustrzane, 643
 - architektura, 644
 - automatyczne przełączanie awaryjne, 657, 660
 - baz danych, 516
 - funkcje, 659
 - danych, 635
 - grupy dostępności, 676
 - klastry, 675
 - przesyłanie dzienników, 676
 - punkty końcowe, 648

kopie lustrzane
 replikacja transakcyjna, 676
 ręczne przełączanie awaryjne, 662
 synchronizacja serwerów, 654
 tryb wysokiego bezpieczeństwa, 656
 tryby działania, 645
 uruchamianie sesji, 655
 używanie monitora, 665
 wymuszone przełączenie awaryjne, 664
 zmiana ról, 659

kopie zapasowe, 551, 827
 indeksów pełnotekstowych, 577
 różnicowe, 554
 różnicowe częściowe, 554

kostka SSAS, 38

koszt stosowania indeksów, 469

kreator

bazy SQL Database, 798–801
 Deployment Wizard, 719–721
 grupy dostępności, 824
 Import and Export Wizard, 557, 701
 konfiguracji bazy MDW, 267
 kopiowania baz danych, 563–569
 New Session Wizard, 391–396
 Package Installer Wizard, 692, 693
 planów konserwacji, 148
 skryptów, 557

kroki zadań, 156, 159

kubełek, 255

kwerendy

o niskiej wydajności, 421
 oczekujące, 407
 usług, 157

kworum, 518, 657

L

licencje, 55

na rdzeń procesora, 55
 na serwer, 56

liczba

dysków, 303
 indeksów, 469
 węzłów, 523

licznik

Bajty prywatne, 368
 Bajty wirtualne, 368
 Buffer Cache Hit Ratio, 368
 CLR, 228
 Czas procesora, 359

Czas uprzywilejowany, 359
 Długość kolejki procesora, 359
 Dostępna pamięć, 367
 Free Pages, 368
 Page Life Expectancy, 368
 zasobów procesora, 359
 Zestaw roboczy, 368

liczniki dysków

fizycznych, 361
 logicznych, 361

lista

baz SQL Database, 802
 dostępnych raportów, 412
 raportów, 421
 SharePointa, 783
 zdarzeń, 398

login, 236, 238, 628

lokalizacja kopii zapasowych, 585, 590

lokalność danych, 343

Ł

łańcuchy uprawnień, 244–247

łączenie operatorów dostępu, 434

M

macierz RAID, 62, 307, 363

sprzętowa, 364

magazyn wersji, 454

maksymalna ilość pamięci, 343

maska koligacji, 336

MAXDOP, 336

mechanizm

klastrowania, 516
 odświeżania danych, 790
 SQL Trace, 375
 szacowania kardynalności, 419

menedżer

pamięci wirtualnej, 298
 zadań, 291

metadane, 843

metody zapewniania wysokiej dostępności, 675

miejsce tworzenia kopii zapasowych, 828

minimalny przestój, 639

model

BULK_LOGGED, 572
 FULL, 572
 MOLAP, 713
 P2P, 481
 SIMPLE, 573

- modele
 - abstrakcyjne, 713
 - odzyskiwania, 570, 572
 - replikacji, 478
 - standardowe, 713
 - tabelowe, 713
 - tabelaryczne SSAS, 38
 - moduł, 382
 - modyfikowanie
 - danych, 440
 - sesji, 399
 - monitor
 - tworzenia kopii lustrzanych, 665
 - wydajności, 291, 356, 358, 508
 - monitorowanie, 340
 - aktywności, 688
 - długości kolejek, 365
 - dysków, 365
 - dziennika błędów, 410
 - dzienników, 410
 - dzienników zdarzeń, 411
 - ilości dostępnej pamięci, 367
 - kompresji danych, 331
 - liczników, 365
 - odczytu i zapisu, 365
 - opóźnień, 364
 - pracy, 350
 - procesów, 136
 - przepustowości, 363, 364
 - replikacji, 506
 - systemu, 353, 400, 625
 - procedury składowane, 626
 - transferu, 365
 - użytkowania pamięci, 367
 - wydajności, 228
 - zdarzeń, 371
 - zdarzeń SSAS, 731
- N**
- narzędzia
 - administracyjne, 114
 - do monitorowania wydajności, 369
 - do rozwiązywania problemów, 121
 - konfiguracyjne, 114
 - Microsoft SQL Server Data Tools, 226
 - monitorujące, 356
 - narzędzie
 - Activity Monitor, 132
 - Activity Monitor w SQL Server Management Studio, 357
 - APS, 29
 - ARM, 249, 266
 - Azure Management Portal, 798, 799
 - BCP, 557
 - Configuration Manager, 734
 - Database Mirroring Monitor, 668
 - Distributed Replay, 378
 - DRU, 105
 - DTA, 468, 469, 470
 - DTEExec, 702
 - DTEExecUI, 702
 - DTUtil, 689, 694
 - Execute Package, 703
 - Failover Cluster Manager, 830
 - Flight Recorder, 731
 - Logman, 369
 - PDW, 29
 - PerformancePoint, 782
 - Relog, 370
 - Report Manager, 741, 747
 - SQL Server 2014 Upgrade Advisor, 98
 - SQL Server Agent, 156, 171, 673, 704
 - SQL Server Backup to URL, 37
 - SQL Server Backup to Windows Azure, 37
 - SQL Server Configuration Manager, 86
 - SQL Server Management Studio, 126
 - SQLDiag.exe, 142
 - SQLDumper.exe, 141
 - SSDT, 701, 719, 738
 - SSDT-BI, 38
 - SSS, 785, 786
 - System Center Advisor, 411
 - WMI, 170
 - natychmiastowe inicjowanie plików, 284
 - nawiązywanie połączenia, 807
 - nazwa FQDN serwera, 808
 - nienadzorowane konto usługowe, 781, 785
 - nieudana instalacja, 88
 - niezależna baza danych, 831
 - niezależny użytkownik, 236
 - niezawodność, 232
 - nowe funkcje, 36, 92
 - nowe subskrypcje, 758
- O**
- obciążenie, 312
 - obiekty
 - DMF, 45
 - DMO, 44
 - SSAS, 721

obiekty
 typu pred_compare, 386
 typu pred_source, 387
 zdarzeń rozszerzonych, 382
 obliczenia, 713
 obraz z kopią zapasową, 577
 obsługa
 błędów, 725
 grup dostępności, 823
 przekierowywania połączeń, 840
 techniczna, 141
 wejścia-wyjścia, 312
 odbieranie komunikatów, 206
 odbiornik
 grupy dostępności, 838
 XMLA, 714
 odświeżanie
 danych, 776, 784–790
 pamięci podręcznej, 765
 odtwarzanie śladu, 377
 odzyskiwanie, 570, 580
 baz danych, 595
 systemu, 609
 ograniczanie dostępu do zasobów, 805
 okno
 Installation Center, 67
 New Alert, 167
 Object Explorer, 225, 813
 Object Explorer Details, 126
 projektanta planów konserwacji, 154
 Ready to Install, 69
 określanie
 celów monitorowania, 354
 punktu odniesienia, 355
 OLAP, Online Analytical Processing, 712
 OLTP, Online Transaction Processing, 36, 249
 opcje
 generowania raportów, 152
 importu i eksportu, 558
 kopii zapasowych, 591
 przetwarzania, 724
 przywracania bazy, 602
 replik, 826
 snapshotów, 767
 sprzętowe, 58
 systemu, 81
 śledzenia, 140, 287
 uruchomieniowe, 117
 uwierzytelniania, 787

operacje
 DBCC, 83
 na indeksach, 452
 na indeksie, 464
 wejścia-wyjścia, 300, 312
 operator awaryjny, 164
 operatory
 dostępu, 427, 434
 sortowania, 83
 złączeń, 435
 operatorzy, 163
 opóźnienia dysków twardych, 303
 opóźnione utrwalanie, 37
 oprogramowanie, 63
 optymalizator kwerend, 110, 419
 optymalizowanie
 aplikacji, 311
 sprzętu serwera, 292
 systemu, 311

P

PaaS, Platform as a Service, 795
 pakiety, 382
 poprawek i aktualizacje, 85
 sieciowe, 83
 SSIS, 38, 568, 688, 694
 pamięć, 59, 82, 258, 297, 339
 bliska, 343
 daleka, 343
 DAS, 61
 DIMM, 297
 dyskowa, 60
 fizyczna, 297
 flash, 305
 NAND, 305
 NOR, 305
 podręczna, 295
 RAM, 297
 zajęta, 409
 panel kontrolny grupy dostępności, 845
 parametry
 uruchomieniowe, 116
 wiersza poleceń, 71
 parsowanie, 417
 partycje, 37, 320, 329, 362, 454, 606
 PDW, Parallel Data Warehouse, 29
 PerformancePoint
 źródła danych, 782
 pętla zagnieżdżona, 435

- PFS, Page Free Space, 318
- plan
 - konserwacji bazy danych, 147, 148, 589
 - odświeżania danych, 792
 - tworzenia kopii zapasowych, 584, 634
 - wykonywania kwerend, 409, 423–426, 437, 440, 443
 - zasilania, 282
- planowanie systemu, 58
- platforma
 - .NET, 221
 - SharePoint 2013, 769
 - Windows Azure, 833
- plik
 - CertSerwerGlowny.sql, 649
 - dostępDoIndeksu1.sql, 428
 - dostępDoIndeksu2.sql, 430
 - dostępDoIndeksu3.sql, 431
 - dostępDoIndeksu4.sql, 433
 - Instalacja-Sql2014.ps1, 73
 - konfiguracyjny, 71
 - logman_create_io.cmd, 370
 - nieuzywaneIndeksy.sql, 407
 - oczekiwanieNaPamiec.sql, 407
 - planyWykonywanychKwerend.sql, 409
 - stronicowania, 299
 - tworzKolejki.sql, 380
 - tworzPowiadomOZdarz.sql, 380
 - uzywaneIndeksy.sql, 405
 - wykonywaneKwerendy.sql, 402
 - wykorzystPamieci.sql, 409
 - wykorzystZasobow.sql, 402
 - XEdlugieKwerendy.sql, 389
 - XEnodyfSesji.sql, 389
 - zlaczanie.sql, 435
 - zlaczanie3.sql, 439
- pliki bazy danych, 39
- podejście równoległe, 97
- podział
 - na partycje, 442, 606
 - tabel i indeksów, 320, 606
- polecenia
 - systemu operacyjnego, 156
 - usług SSAS, 157
- połączenia
 - aplikacji klienckich, 837
 - zarządzane, 780
- połączenie
 - DAC, 121
 - z bazą master, 809
 - ze źródłem danych, 777
- pomiar obciążenia pamięci, 341
- poprawki SP, 544
- porty TCP/IP, 85
- porządkowanie indeksów, 463
- POS, Point of Service, 480
- Power View, 775
- PowerPivot, 770, 783
 - odświeżanie danych, 785
- PowerPivot Data Refresh, 792
- powiadomienia
 - dla kwerend, 83
 - o zadaniach, 161
 - o zdarzeniach, 83, 372, 378
 - operatorów, 164
- poziomy ochrony pakietu, 707
- predykat, 386
- priorityty, 201
- problemy
 - z aktualizacją, 102
 - z klastrami, 547
- procedura
 - sp_configure, 130
 - sp_replcounters, 509
 - sp_who, 137
- procedury składowane, 243, 263
 - natywnie kompilowane, 263
- proces
 - przesyłania dzienników, 613
 - przetwarzania kwerend, 417
 - przywracania, 595
- procesor, 58, 258, 294, 333
 - czas, 359
 - kwerend, 714
- procesy
 - BI, 279
 - ETL, 38
- produkcyjne bazy danych, 36, 291
- projektant planów konserwacji, 153
- przebieg szeregowania, 336
- przechowywanie
 - danych, 305, 614
 - połączeń, 781
- przekazywanie zdarzeń, 183
- przekierowywanie
 - klientów, 633, 674
 - operacji odczytu, 840
 - połączeń, 840
- przełączenie
 - awaryjne, 520, 670, 820, 834
 - automatyczne, 660, 820
 - nieplanowane, 632

przełączanie
 awaryjne
 planowane, 630
 ręczne, 662, 820
 węzłów, 545
 wymuszone, 664
 kontekstu, 335
 między serwerami, 632
 modeli odzyskiwania, 573
 ról, 630
 przenoszenie
 grup, 544
 loginów, 628
 przestawianie partycji, 329
 przestój, 672
 przestój standardowy, 639
 przesyłanie dzienników, 515, 607–610, 676
 aktualizacja, 639
 kończenie, 637
 scenariusze, 608
 wdrażanie, 615
 wydajność, 638
 wymagania systemowe, 613
 przetwarzanie
 kostek, 723
 kwerend, 416, 442
 obiektów SSAS, 721
 partycji, 723
 ponowne, 723
 w modelu MOLAP, 722
 w modelu tabelowym, 725
 wymiarów, 722
 przypisywanie
 ról użytkownikom, 749
 uprawnień, 811
 przyspieszanie zapisu, 304
 przystawka Usługi systemu Windows, 685
 przywracanie, 578
 bazy danych, 596, 597, 729
 dzienników transakcji, 596
 grup plików, 597
 kopii zapasowych, 727
 plików, 597
 snapshotów, 598
 stanu, 551, 553, 575
 systemowych baz danych, 604
 tabel metadanych, 599
 publikacja ze snapshotem, 493

pula
 BPE, 339, 340
 buforów, 409
 zasobów, 345, 348
 punkty końcowe, 238

R

RAID, 62, 307, 363
 RAID 0, 307
 RAID 1, 307
 RAID 10, 308
 RAID 5, 308, 595
 RAID 6, 308
 raport
 Contention Analysis, 276
 Stored Procedure Based, 277
 Usage Analysis, 274, 275
 raporty, 124
 o blokadach, 136
 o bazach danych, 124
 o błędach, 88
 standardowe, 411
 z poziomu serwera, 124
 rdzeń, 296
 reagowanie na alerty, 170
 reindeksacja, 83
 rejestrowanie
 danych, 159
 danych o krokach zadań, 159
 śladu, 378
 rekordy, 252
 relacje z klientami, 480
 replika pomocnicza
 dostęp tylko do odczytu, 839
 kopie zapasowe, 842
 połączenie, 840
 wydajność, 841
 replikacja, 473, 515, 636
 agenty, 475
 dane replikacji, 475
 implementowanie, 483
 migawkowa, 476, 499
 modele, 478
 monitor, 506
 P2P, 502
 procedura sp_replcounters, 509
 role, 474
 skrypty, 505
 transakcyjna, 477, 676

- transakcyjna ze skalaniem, 500
 - w trybie P2P, 477, 501
 - widoki DMV, 508
 - z publikacją, 478
 - zadania, 476
 - ze skalaniem, 477
 - repliki, 819, 831
 - repliki pomocnicze, 821, 839
 - Report Manager, 747
 - historia raportu, 766
 - menu nawigacyjne, 748
 - menu zadań, 750
 - New Subscription, 757
 - opcje przetwarzania, 764
 - strona Data Sources, 756
 - zabezpieczenia, 767
 - zarządzanie raportami, 754
 - Resource Governor, 37, 344
 - grupy operacyjne, 346
 - klasyfikowanie, 347
 - monitorowanie pracy, 350
 - pule zasobów, 345
 - rodzaje
 - indeksów, 458
 - klastrów, 522
 - pamięci flash, 305
 - uwierzytelniania, 234
 - rola
 - SQLAgentReaderRole, 172
 - SQLAgentUserRole, 172
 - role
 - bazy danych, 575, 708
 - serwera, 575
 - użytkownika, 241
 - wbudowane, 242
 - w grupach dostępności, 819
 - rozmieszczanie plików, 313
 - rozszerzenia puli buforów, 339
 - rozszerzenie Power View, 775
 - rozwiązywanie problemów, 121
 - równoległe przetwarzanie kwerend, 321
 - równoważenie obciążenia, 805
 - równoważenie obciążenia sieci, 633
 - różnicowe kopie zapasowe plików, 554
 - rysunki VDW, 784
- S**
- SAN, Storage Area Network, 61
 - SAS, Serial Attached SCSI, 303
 - SCA, System Center Advisor, 411
 - scalanie partycji, 329
 - schematy, 43
 - bazy danych, 290
 - partycjonowania, 324
 - SCSI, Small Computer Systems Interface, 303
 - Secure Store Service, 779
 - Server Objects, 813
 - Service Broker, 189, 191, 192
 - grupy konwersacji, 202
 - kolejki, 197
 - kontrakty, 196
 - odbieranie komunikatów, 206
 - priorytety, 201
 - przesyłanie komunikatów, 207
 - punkty końcowe, 209
 - trasy, 199
 - typy komunikatów, 195
 - usługi, 199
 - ustawianie stanu, 193
 - wysyłanie komunikatów, 203
 - zapisywanie danych, 211
 - zewnętrzna aktywacja, 209
 - serwer
 - docelowy, 185
 - DNS, 633
 - główny, 612
 - lustrzany, 670, 673
 - monitorujący, 612, 623
 - nadrzędny, 185
 - pomocniczy, 612
 - rezerwowany, 608
 - SQL Database, 799, 801
 - SSAS, 711, 715
 - elementy architektury, 714
 - instalowanie baz, 719
 - język ASSL, 717
 - właściwości, 715
 - wymagane usługi, 717
 - zarządzanie, 715
 - zarządzanie bazami, 719
 - SSRS, 747
 - świadka, 657
 - usług SSIS, 685
 - zapasowy w trybie offline, 514
 - zapasowy w trybie online, 515
 - sesja
 - system_health, 374
 - zdarzeń, 388
 - zdarzeń rozszerzonych, 391
 - SGAM, Shared Global Allocation Map, 318

- sieci, 301
 - prywatne, 519
 - publiczne, 519
 - SAN, 61
- silnik
 - bazodanowy, 113
 - OLAP, 712
 - RDBMS, 712
- skalowalność, 231
- skalowanie systemu, 745
- skanowanie
 - indeksów klastrowanych, 429
 - indeksów nieklastrowanych, 432
 - tabeli, 428
- składnia operacji na indeksie, 464
- skrypty, 157
- skrypty powłoki PowerShell, 72
- snapshot, 490, 493, 598, 677, 767
- sortowanie, 64
- spójność pamięci podręcznej, 334
- sprawdzanie
 - połączeń, 135
 - poprawności obrazów, 577
 - poprawności replikacji migawkowej, 499
 - systemu, 81
 - węzłów, 532
- sprzęt serwera, 292
- SQL Database
 - administrowanie, 795, 809
 - konfigurowanie, 798
 - kopie zapasowe, 813
 - nawiązywanie połączenia, 807
 - niedostępność narzędzi, 815
 - replikacja, 814
 - warstwa infrastruktury, 797
 - warstwa klientów, 797
 - warstwa platformy, 797
 - warstwa usług, 797
- SQL Server 2014, 29, 35
- SQL Server Agent, 704, 815
 - jednostki pośredniczące, 173
 - kategoria Advanced, 177
 - kategoria Alert System, 178
 - kategoria Connection, 180
 - kategoria General, 176
 - kategoria History, 180
 - kategoria Job System, 179
 - konfiguracja, 176
 - konfigurowanie zadań, 673
 - podsystemy, 173
 - zabezpieczenia, 171
 - zadania, 156, 630
- SQL Server Configuration Manager, 114, 686
 - właściwości usługi, 686
- SQL Server Data Tools, 226
- SQL Server Management Studio, 123, 126, 292, 584, 599, 808
 - ekran Server Properties, 126
 - konfigurowanie systemu, 126
 - kończenie przesyłania dzienników, 637
 - monitorowanie systemu, 626
 - Object Explorer Details, 126
 - przesyłanie dzienników, 616
 - raporty, 124
 - zakładka Advanced, 129
 - zakładka Connections, 129
 - zakładka Database Setting, 129
 - zakładka Memory, 127
 - zakładka Permissions, 130
 - zakładka Processors, 127
 - zakładka Security, 129
 - zarządzanie pakietami, 689
- SQL Server Native Client, 40
- SQL Server Profiler, 356, 371, 375
- SQL Server Upgrade Advisor, 98
- SQL Trace, 357, 371, 374
- SQLAgentOperatorRole, 172
- SQLCLR, 219
- SQLOS, 342
- SSAS, SQL Server Analysis Services, 679
- SSD, Solid State Drive, 62
- SSDT, SQL Server Data Tools, 683
- SSRS, SQL Server Reporting Services, 679
- standardowe raporty, 357, 413
- statystyki oczekiwania, 404
- stosowanie klastrowania, 514
- strony, 318
- subskrybent, 475, 478, 480
- subskrybent aktualizujący dane, 481
- subskrypcja, 764
- subskrypcja sterowana danymi, 759
- synchronizowanie
 - baz SSAS, 729
 - loginów, 236
 - serwerów, 654
 - zależności, 627
- synonimy, 44
- system
 - Center Advisor, 357
 - SQLOS, 342
 - wielordzeniowy, 296
 - zgłaszania problemów, 322

systemowe
 bazy danych, 41
 procedury składowane, 357
 szablon sesji zdarzeń, 393
 szacowanie
 kardynalności, 419
 oszczędności miejsca, 329
 szeregowanie zadań, 334
 szyfrowane kopie zapasowe, 37, 553

Ś

śledzenie, 340
 środowisko
 CLR, 52, 219–222
 produkcyjne, 444
 SQLCLR, 815
 systemu, 35
 Visual Studio, 224
 wirtualne, 56

T

tabele
 metadanych, 599
 podzielone na partycje, 454
 z historią kopii, 574
 zoptymalizowane, 250, 260
 tabelowe źródła danych, 782
 technologia
 AlwaysOn, 817
 Hyper-Threading, 295
 In-Memory OLTP, 249
 Resource Governor, 345
 SQLCLR, 221
 testy, 544
 po aktualizacji, 110
 wydajności, 284, 425
 tokeny, 181
 transakcje OLTP, 36
 trasy, 199
 tryb
 DirectQuery, 79
 DirectQuery with In-Memory, 79
 EXECUTE AS OWNER, 264
 HOLAP, 78
 In-Memory, 79
 In-Memory with DirectQuery, 79
 MOLAP, 78, 712

NORECOVERY, 677
 ROLAP, 78
 tabelaryczny, 78
 tylko do odczytu, 821, 839
 UDM, 78
 uprzywilejowany, 335
 tryby
 dostępności, 819
 działania kopii lustrzanych, 645
 kworum, 519
 przełączania awaryjnego, 820
 wysokiego bezpieczeństwa, 656
 wysokiej wydajności, 658
 tworzenie
 aplikacji usługowej, 774
 certyfikatów, 208
 funkcji partycjonującej, 322
 grup plików, 324
 grupy zasobów, 540
 indeksów, 459
 klastra, 525, 531
 komponentów CLR, 223
 konta użytkownika, 811
 kopii lustrzanych, 648, 653, 669, 838
 kopii zapasowej, 86, 154, 552, 575, 728, 813, 842
 loginu, 810
 obciążenia, 372
 planu, 153
 punktu odniesienia, 372
 schematu partycjonowania, 324
 serwera rezerwowego, 608
 SQL Database, 798
 subskrypcji, 494
 tabeli, 813
 typy
 danych, 45, 47
 binarne, 49
 dokładne liczbowe, 48
 przybliżone liczbowe, 48
 systemowe, 51
 z datą i czasem, 49
 znakowe, 47
 kompresji, 330
 komunikatów, 195
 LOB, 83
 przełączania awaryjnego, 820
 publikacji, 488
 replikacji, 476
 serwerów SSAS, 715

U

umożliwianie aktualizacji skoroszytu, 790

uprawnienia, 575

- do bazy danych, 243
- do tabel i widoków, 243
- na serwerze, 237
- podsystemów, 173
- z poziomu egzemplarza, 239, 240, 241

uruchamianie

- narzędzia DRU, 105
- narzędzia Upgrade Advisor, 101
- pakietów
 - język T-SQL, 706
 - kreator, 701
 - narzędzie DTEExec, 702
 - narzędzie DTEExecUI, 702
 - narzędzie Execute Package, 703
 - narzędzie SSDT, 701
- przetwarzania, 723

uruchomieniowe procedury składowane, 118

usługa, 65, 199

- SSIS, 29, 38
- SSRS, 38
- typu PaaS, 795

usługi

- Analysis Services, 76, 87, 711, 783
- Excela, 779, 783
- Integration Services, 679
- PerformancePoint, 783
- Reporting Services, 87, 733, 772
- SSAS, 711
 - dostrajanie wydajności, 730
 - monitorowanie, 730
- SSIS, 680
 - główne elementy, 681
 - komponenty przepływu danych, 682
 - komponenty wykonawcze, 682
 - konfiguracja, 684
 - serwer, 685
 - właściwości, 685
 - zabezpieczenia, 706
 - zarządzanie, 683
 - zastosowania, 680
- SSRS
 - Configuration Manager, 734
 - konto usługowe, 736
 - Report Manager, 741
- Visio, 784
 - metody uwierzytelniania, 784
 - odświeżanie danych, 784

- WMI, 184
- usprawnienia widoków indeksowanych, 454
- ustawianie
 - alertów, 668
 - katalogu, 753
 - modelu odzyskiwania, 616
 - nowej sesji, 399
 - pamięci, 82
 - planu zasilania, 283
 - poczty elektronicznej, 742
 - replik pomocniczych, 822, 826
- ustawienie
 - Cost Threshold for Parallelism, 337
 - MAXDOP, 336
 - SAFETY FULL, 672
 - SAFETY OFF, 673
- usuwanie
 - baz danych, 834
 - dużych indeksów, 453
 - instalacji silnika bazodanowego, 88
 - instalacji systemu, 87
 - instalacji usług, 87
 - replik, 831
- uwierzytelnianie, 234
 - dostępu, 778
 - w SQL Server, 234
 - w Windows, 235
- użytkowanie pamięci, 366
- użytkownik, 236, 289, 628
- używanie
 - indeksów, 405, 461
 - narzędzia Upgrade Advisor, 99
 - programu SQLDiag.exe, 144
 - Service Brokera, 203
 - sesji zdarzeń rozszerzonych, 391
 - tabel, 461
 - technologii Resource Governor, 349
 - usług WMI, 184

V

Visual Studio, 224

VMM, Virtual Memory Manager, 298

W

warstwa

- infrastruktury, 797
- klientów, 797
- platformy, 797

- prezentacji, 776
 - usług, 797
 - wątek, 296
 - wątek lekki, 335
 - wdrażanie przesyłania dzienników, 616, 624
 - wersje kodu systemu, 93
 - węzły, 504
 - aktywne, 517
 - pasywne, 517
 - widok
 - DASHBOARD, 803
 - sys.dm_exec_connections, 137
 - sys.dm_xe_packages, 383
 - źródła danych, 713
 - widoki
 - DMV, 231, 263, 316, 400, 846
 - systemowe, 445
 - podzielone na partycje, 607
 - z katalogu, 390
 - wielkość znaków, 64
 - wiersz poleceń, 70
 - Windows Azure, 37
 - Windows Azure SQL Database, 795
 - Windows System Monitor, 228
 - wirtualizacja, 258
 - właściwości
 - dotyczące pamięci, 716
 - dzienników, 716
 - OLAP, 716
 - raportu, 755
 - serwera SSAS, 715
 - sieci, 716
 - zabezpieczeń, 497
 - zadania, 155
 - związane z bezpieczeństwem, 717
 - włączanie
 - In-Memory OLTP, 250
 - parametrów uruchomieniowych, 116
 - WMI, Windows Management Instrumentation, 170
 - wolna przestrzeń, 408
 - współbieżne wykonywanie kwerend, 444
 - współużytkowana macierz dyskowa, 517
 - wybieranie
 - modelu odzyskiwania, 572
 - modelu subskrypcji, 495
 - pamięci, 61
 - publikacji, 495
 - publikowanych tabel, 489
 - raportów, 274
 - trybu usług, 77
 - wydajność, 168, 228, 231, 289, 291
 - dysku, 366
 - kodu, 415
 - kwerend, 420, 444, 464
 - przesyłania dzienników, 638
 - serwera, 279
 - wydawca, 474, 478
 - wykonywanie
 - kwerendy, 401
 - pakietów usług SSIS, 157
 - wykorzystanie liczników pamięci, 369
 - wyłączanie zegara, 793
 - wymagania sprzętowe, 59
 - wymiana certyfikatów, 208
 - wyrażenia CTE, 83
 - wysoka dostępność, 583, 635
 - wysyłanie
 - alertów, 668
 - komunikatów, 203
 - wyszukiwanie serwera wydawcy, 494
 - wyświetlanie kwerend, 407
 - wyzwalacze, 83
 - wznawianie pracy bazy danych, 837
 - wzorce użytkowania danych, 290, 581
- X**
- XLMA, 714
- Z**
- zabezpieczanie
 - egzemplarzy baz danych, 233
 - połączeń, 781
 - usług SSIS, 706
 - zabezpieczenia, 171
 - agenta dystrybucji, 496
 - na poziomie wierszy, 247
 - przy integracji, 227
 - zadania, 156
 - zadania powłoki PowerShell, 157
 - zakleszczenie, 372
 - zależności baz danych, 630
 - zapis
 - na adres URL, 552
 - w dzienniku, 211
 - w Windows Azure, 552
 - zapisywanie środowiska, 104
 - zapora, 804
 - zapora baz SQL Database, 806

zarządzane tworzenie kopii, 552

zarządzanie

bazami, 719

klastrem, 546

kluczami szyfrowania, 744

kopiami zapasowymi, 594

pakietami, 689

pakietami SSIS, 688, 694

pamięcią, 261

połączeniami, 781

projektem, 683

raportami, 754

relacjami z klientami, 480

serwerem SSAS, 715

silnikiem bazodanowym, 113

sprzętem, 293

usługami

Analysis Services, 711

Integration Services, 679

Reporting Services, 733

SIS, 683

użytkownikami, 749

wieloma serwerami, 138, 180, 185

zmianą ról, 627

zasady zapory, 804

zatory w pamięci, 369

zaufane

biblioteki połączeń, 780

lokalizacje plików, 780

zaufani dostawcy danych, 781

zawieszanie pracy bazy danych, 836

zdarzenia, 170, 183, 383

rozszerzone, 292, 356, 371, 381, 385

rozszerzone, 230

SSAS, 731

systemu, 166

zgłaszanie problemów, 322

zgodność wstecz, 106

złączanie

przez scalanie, 438

w pętli, 435

z haszowaniem, 437

zmiana ról, 627

Ż

źródła danych, 756

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

Podręcznik administratora Microsoft SQL Server 2014 zawsze pod ręką!

Microsoft SQL Server to jedna z najczęściej wybieranych platform bazodanowych. W jej tabelach przechowywane są niewyobrażalne ilości danych, a zdolność do ich szybkiego przetwarzania decyduje o byciu albo nie byciu wielu firm. Sięgnij po tę książkę i zapewnij Twoim zbiorom danych najwyższą wydajność, niezawodność oraz bezpieczeństwo.

Została ona w całości poświęcona zagadnieniom związanym z administrowaniem najnowszą wersją Microsoft SQL Server. Znajdziesz tu informacje związane z instalacją, wymaganiami sprzętowymi oraz wstępną konfiguracją. Ponadto poznasz najlepsze praktyki w zakresie przeprowadzania aktualizacji, zarządzania silnikiem bazodanowym oraz monitorowania wydajności. W kolejnych rozdziałach zawarto bezcenną wiedzę na temat automatyzacji zadań, integracji z innymi usługami, zabezpieczania danych oraz optymalizacji działania systemu. Książka porusza również zaawansowane zagadnienia związane z klastrowaniem w systemie SQL Server 2014 oraz z zarządzaniem usługami i grupami dostępności AlwaysOn. Jest ona kompletnym, najlepszym źródłem wiedzy dla każdego administratora, którego codzienne obowiązki są związane z systemem bazodanowym Microsoft SQL Server 2014.

Dzięki tej książce:

- zainstalujesz platformę Microsoft SQL Server 2014
- dobierzesz odpowiedni sprzęt
- zoptymalizujesz wydajność Twojej bazy danych
- wykonasz backup danych i poznasz procedurę przywracania z kopii bezpieczeństwa
- będziesz mieć pod ręką kompletne źródło wiedzy na temat Microsoft SQL Server 2014

Adam Jorgensen — jest prezesem firmy Pragmatic Works i wiceprezesem organizacji PASS. Od ponad 15 lat jest związany z branżą IT i systemami bazodanowymi.

Bradley Ball — otrzymał certyfikat MCITP w 2005 i 2008 roku, a MCSE DBA w 2012 roku. Jest kierownikiem ds. zarządzania platformą danych w firmie Pragmatic Works. Specjalizuje się w tworzeniu rozwiązań opartych na tej platformie. Posiada bogate, ponad dziesięcioletnie doświadczenie w tej dziedzinie.

Steven Wort — aktywnie używa systemu SQL Server od 1993 roku. W 2000 roku dołączył do zespołu Microsoftu. Aktualnie zajmuje stanowisko architekta w zespole pracującym nad usługami CRM.

Ross LoForte — jest architektem technologii w centrum technologicznym firmy Microsoft w Chicago. Swoje działania koncentruje na rozwiązaniach opartych na platformie SQL Server. Od ponad 20 lat zajmuje się rozwijaniem oprogramowania, zarządzaniem projektami oraz tworzeniem zaawansowanych rozwiązań w języku SQL.

Brian Knight — jest założycielem Pragmatic Works, a także współzałożycielem serwisów SQLServerCentral.com oraz JumpStart TV. Jest autorem lub współautorem ponad 15 książek. Występuje jako prelegent na licznych konferencjach.

sięgnij po WIĘCEJ



KOD KORZYSCI

Helion

35063 numer katalogowy

księgarnia internetowa

<http://helion.pl>

zamówienia telefoniczne

☎ 0 801 339900

☎ 0 601 339900

Informatyka w najlepszym wydaniu

Sprawdź najnowsze promocje:
• <http://helion.pl/promocje>
Książki najchętniej czytane:
• <http://helion.pl/bestsellery>
Zamów informacje o nowościach:
• <http://helion.pl/nowości>

Helion SA
ul. Kościuski 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
<http://helion.pl>

ISBN 978-83-283-0673-8



9 788328 306738

cena: 129,00 zł

Wrox
An Imprint of
WILEY