OCHRONA TWOICH DANYCH W EPOCE TERABAJTÓW

Profesjonalne tworzenie kopii zapasowych i odzyskiwanie danych

Steven Nelson

Apress

Helion

Tytuł oryginału: Pro Data Backup and Recovery

Tłumaczenie: Grzegorz Kowalczyk (wstęp, rozdz. 1, 3 – 11), Witold Wrotek (rozdz. 2)

ISBN: 978-83-246-3417-0

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)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/prokop
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

ROZDZIAŁ 5.

Nowe technologie nośników

Przez wiele lat tworzenie kopii zapasowych było zajęciem rutynowym — najpierw należało przenieść dane na taśmę magnetyczną, a następnie na wszelki wypadek utworzyć dodatkowo kopię takiej taśmy. Po pewnym czasie na rynku pojawiły się biblioteki VTL, które spowodowały, że tworzenie kopii zapasowych stało się trochę mniej nudnym zajęciem, aczkolwiek nadal stanowiło tylko pewną modyfikację pierwowzoru. W ostatnich latach jednak pojawiło się kilka bardzo ciekawych technologii i metod, które wniosły nowe tchnienie w nieco już zastały świat kopii zapasowych. Co więcej, nowe platformy i systemy operacyjne, a zwłaszcza szeroko wprowadzana wirtualizacja serwerów, przyniosły nowe sposoby tworzenia kopii zapasowych, które nie mieszczą się w ich tradycyjnej definicji. W tym rozdziale omówimy nowe technologie, takie jak deduplikacja, ciągła ochrona danych i aplikacji zdalnych, VMWare Recovery API oraz przetwarzanie w chmurze.

Deduplikacja

Jedną z najczęściej dziś omawianych nowych technologii jest **deduplikacja** (ang. *deduplication*), czyli proces analizy danych na poziomie podplikowym (na poziomie bloków danych) i zapisywanie tylko tych elementów, które do tej pory nie zostały zapisane w pamięci masowej. W niektórych definicjach deduplikacji analizy danych dokonuje się tylko na poziomie pliku, dzięki czemu pojedynczy plik występujący w wielu lokalizacjach (na przykład dokument edytora Word) jest zapisywany w kopii zapasowej tylko raz. Tak naprawdę jednak nie jest to pełna deduplikacja, a tylko proces nazywany **SIS** (ang. *Single Instance Storage*; przechowywanie tylko jednego egzemplarza obiektu). Podczas pełnej deduplikacji dane są odczytywane, dzielone na bloki i porównywane z blokami, które zostały już wcześniej zapisane w pamięci masowej. Jeżeli dany blok nie zostanie odnaleziony, zostaje zapisany w pamięci masowej i od tej pory bierze udział w kolejnych porównaniach. Odnalezione bloki danych są zapisywane jako wskaźniki do istniejących bloków. Dzięki takiemu podejściu ilość rzeczywiście zapisywanych danych zostaje zmniejszona, ponieważ pojedynczy blok danych jest zapisywany tylko raz. Stopień redukcji ilości zapisywanych danych jest mierzalny, ale nie wyrażamy go w ilości zapisywanych danych, ale jako stosunek danych przetwarzanych do danych zapisywanych — wartość ta nosi nazwę **współczynnika deduplikacji** (ang. *deduplication ratio*). Typowa, dobra wartość tego współczynnika wynosi 10:1, a w wielu przypadkach może regularnie osiągać nawet wartości rzędu 15:1 i więcej.

Kilka słów na temat współczynnika deduplikacji (czasami używa się również określenia **współczynnik redukcji danych**; ang. *data reduction ratio*): z matematycznego punktu widzenia współczynnik deduplikacji jest prostą odwrotnością procentowej wartości redukcji danych, jak to zostało przedstawione poniżej:

$$DD = \frac{1}{1 - \%redukcji\_danych}$$

Na przykład: jeżeli wielkość kopii zapasowej dzięki zastosowaniu deduplikacji została zredukowana o 85%, wartość współczynnika deduplikacji będzie następująca:

$$DD = \frac{1}{1 - 0{,}85}$$

$$DD = \frac{1}{0{,}15}$$

$$DD = 6{,}66 \text{ lub } 6:1$$

Odwrotnie, jeżeli wartość współczynnika deduplikacji jest znana, można obliczyć, o ile został zredukowany rozmiar oryginalnej kopii zapasowej. Wartość ta pozwala się zorientować, ile miejsca mogą zająć dane z takiej kopii po odzyskaniu. Aby oszacować procentową wartość redukcji danych, możemy się posłużyć następującym równaniem:

$$\%redukcji\,danych = \left(1 - \left(\frac{1}{DD}\right)\right) * 100$$

Załóżmy, że dostawca rozwiązania deklaruje, że dany produkt posiada współczynnik deduplikacji rzędu 12:1. O ile zostanie zredukowany rozmiar utworzonej kopii zapasowej? Odpowiedź przyniesie podstawienie danych do przedstawionego wcześniej wzoru:

$$\%redukcji\,danych = \left(1 - \left(\frac{1}{12}\right)\right) * 100$$

$$\%redukcji\,danych = \left(1 - 0{,}083\right) * 100$$

$$\%redukcji\,danych = 0{,}916 * 100$$

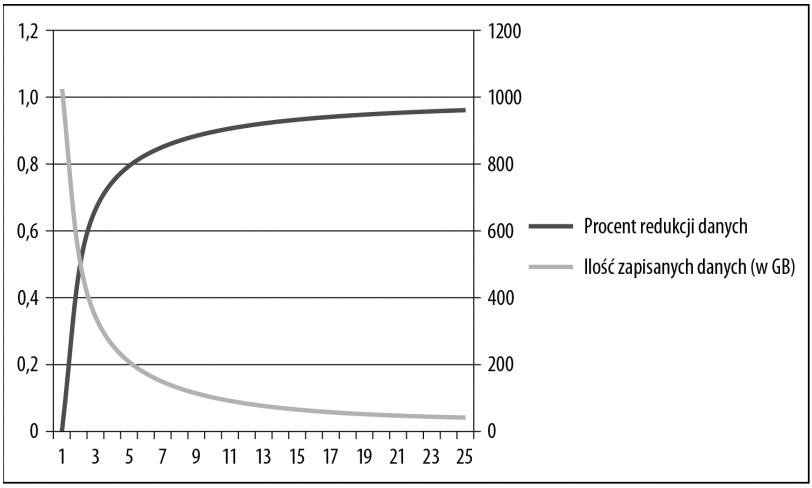
$$\%redukcji\,danych = 91{,}6\%$$

Ale w jaki sposób możemy wyznaczyć najpierw sam współczynnik deduplikacji? Można to zrobić, biorąc zestaw danych o znanym rozmiarze, poddać procesowi deduplikacji i sprawdzić rozmiar zapisanego zestawu danych. Współczynnik deduplikacji DD możemy wyznaczyć, dzieląc rozmiar skopiowanych danych przez rozmiar danych zapisanych w pamięci masowej, tak jak to ilustrujemy poniżej:

$$DD = \frac{rozmiar\,skopiowanych\;danych}{rozmiar\,zapisanych\;danych}$$

Współczynnik deduplikacji, podobnie jak wiele innych statystyk, może być łatwo źle rozumiany lub przyczynić się do błędnej interpretacji użyteczności całego procesu deduplikacji. Wiele produktów wykorzystujących technologię deduplikacji jest reklamowanych przez dostawców jako osiągające współczynniki deduplikacji rzędu 10:1, 15:1, 20:1 czy nawet wyższe. Ciągle rosnące wartości współczynnika deduplikacji mogą sugerować, że

różnica pomiędzy współczynnikiem o wartości 10:1 a 20:1 będzie odpowiadała ogromnej różnicy w ilości danych zapisanych w pamięci masowej. Na pierwszy rzut oka to może tak właśnie wyglądać, ale przyjrzyjmy się rysunkowi 5.1, ilustrującemu wpływ współczynników deduplikacji na ilość zapisywanych danych dla kopii zapasowej o rozmiarze 1 TB.

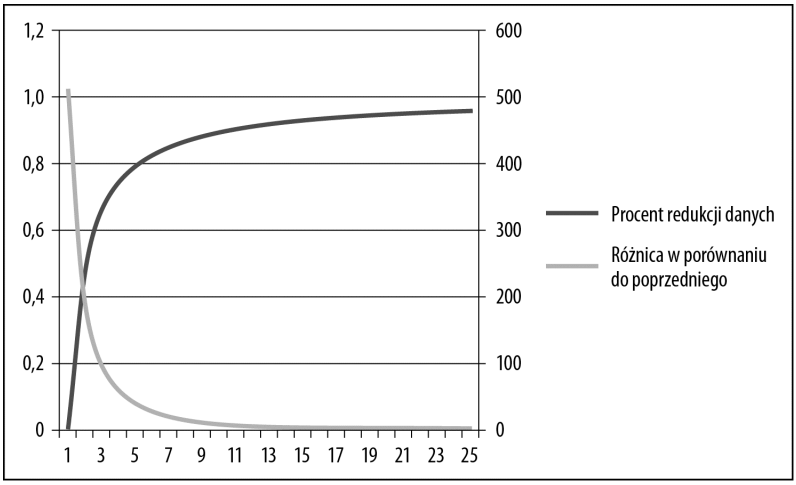


Współczynnik deduplikacji	Procent redukcji danych	Ilość zapisanych danych (w GB)	Współczynnik deduplikacji	Procent redukcji danych	Ilość zapisanych danych (w GB)
1:1	0,00%	1024,00	13:1	92,31%	78,77
2:1	50,00%	512,00	14:1	92,86%	73,14
3:1	66,67%	341,33	15:1	93,33%	68,27
4:1	75,00%	256,00	16:1	93,75%	64,00
5:1	80,00%	204,80	17:1	94,12%	60,24
6:1	83,33%	170,67	18:1	94,44%	56,89
7:1	85,71%	146,29	19:1	94,74%	53,89
8:1	87,50%	128,00	20:1	95,00%	51,20
9:1	88,89%	113,78	21:1	95,24%	48,76
10:1	90,00%	102,40	22:1	95,45%	46,55
11:1	90,91%	93,09	23:1	95,65%	44,52
12:1	91,67%	85,33	24:1	95,83%	42,67
			25:1	96,00%	40,96

Rysunek 5.1. Wpływ wartości współczynnika deduplikacji na ilość zapisanych danych

Wykres i powiązana z nim tabela przedstawione na rysunku 5.1 pokazują relacje pomiędzy wartością współczynnika deduplikacji, ilością danych zapisanych w pamięci masowej (dla kopii zapasowej o rozmiarze 1 TB) oraz wyrażonym w procentach współczynnikiem redukcji danych. W miarę wzrostu wartości współczynnika deduplikacji ilość danych zapisywanych w pamięci masowej maleje, aczkolwiek bardziej interesujące są różnice pomiędzy

wartościami współczynnika deduplikacji w porównaniu z różnicami w ilości zapisywanych danych dla tych współczynników. Kiedy współczynnik deduplikacji osiąga coraz wyższe wartości, różnice w ilości zapisywanych danych są coraz mniejsze — co jest zachowaniem mniej lub bardziej zgodnym z naszymi oczekiwaniami. Co interesujące, różnica w ilości zapisywanych danych dla dwóch sąsiednich wartości współczynnika deduplikacji maleje w miarę wzrostu wartości tych współczynników. Zjawisko to zostało przedstawione na rysunku 5.2.



Współczynnik deduplikacji	Ilość zapisanych danych (w GB)	Różnica	Współczynnik deduplikacji	Ilość zapisanych danych (w GB)	Różnica
1:1	1024,00	0,00	13:1	78,77	6,56
2:1	512,00	512,00	14:1	73,14	5,63
3:1	341,33	170,676	15:1	68,27	4,88
4:1	256,00	85,33	16:1	64,00	4,27
5:1	204,80	51,20	17:1	60,24	3,76
6:1	170,67	34,13	18:1	56,89	3,35
7:1	146,29	24,38	19:1	53,89	2,99
8:1	128,00	18,29	20:1	51,20	2,69
9:1	113,78	14,22	21:1	48,76	2,44
10:1	102,40	11,38	22:1	46,55	2,22
11:1	93,09	9,31	23:1	44,52	2,02
12:1	85,33	7,76	24:1	42,67	1,86
			25:1	40,96	1,71

Rysunek 5.2. Wpływ wartości współczynnika deduplikacji na różnice pomiędzy ilościami zapisanych danych (dla kopii zapasowej o rozmiarze 1 TB)

W miarę wzrostu wartości współczynnika deduplikacji różnica w ilości zapisywanych danych, jak to zostało przedstawione w kolumnie *Różnica*, gwałtownie maleje, zwłaszcza w porównaniu z oryginalnym rozmiarem kopii zapasowej. Wniosek? Porównując różne

produkty na bazie osiąganych wartości współczynnika deduplikacji, warto pamiętać, że większe wartości tego współczynnika nie zawsze oznaczają, że różnice osiągów pomiędzy tymi produktami są znaczące — po prostu ocena osiąganych wartości współczynnika deduplikacji nie jest najlepszą metodą porównywania produktów. Na przykład: jeżeli zgodnie ze specyfikacją produkt A osiąga współczynnik deduplikacji rzędu 20:1, a produkt B jest reklamowany jako osiągający współczynnik deduplikacji na poziomie 25:1, to dla kopii zapasowej o rozmiarze 1 TB różnica w ilości zapisywanych w pamięci masowej danych wynosi tylko 10 GB — *czyli mniej niż 1% oryginalnego rozmiaru 1 TB kopii zapasowej!*

Świetnie — deduplikacja umożliwia zatem zredukowanie ilości zapisywanych w pamięci masowej danych, ale co to w praktyce oznacza i jak ten proces działa? Aby to zilustrować, posłużymy się nieco nietypowym przykładem — jeżeli musiałbyś wykonać kopię zapasową roju miododajnych pszczół, to jaki byłby najbardziej efektywny sposób wykonania takiego zadania?

Pierwszą operacją, jaką należałoby wykonać, byłoby zidentyfikowanie podstawowych elementów składowych pojedynczej pszczoły — takich jak skrzydła, nogi, czarne paski, żółte paski, korpus i głowa. Teraz powinieneś wykonać kopię zidentyfikowanych elementów pierwszej pszczoły — będzie to pierwszy zapisany zestaw elementów składowych pszczoły. Każdy taki element składowy będziemy w terminologii deduplikacji nazywali **blokiem**. Idąc dalej, pszczoły, jak wszystko inne, nie są identyczne — poszczególne pszczoły mają różną liczbę pasków, skrzydełka o różnej długości i rozmiarze i tak dalej, stąd w trakcie przetwarzania i tworzenia kopii zapasowych kolejnych pszczół będziemy zapisywali tylko takie elementy nowej pszczoły, które różnią się od pozostałych, a wszystkie inne standardowe elementy, które zostały zapisane przy okazji przetwarzania poprzednich pszczół, będą „pomijane".

Teraz dodajmy jeszcze inne gatunki owadów pszczołowatych — powiedzmy, że oprócz pszczół miodnych będziemy jeszcze mieli trzmiele. Ze strukturalnego punktu widzenia oba gatunki owadów są takie same, zwiększa się tylko liczba różniących je elementów. Istnieją również pewne oczywiste podobieństwa — na przykład żółte i czarne paski. W metodologii deduplikacji w kopii zapasowej zapisane zostałyby tylko elementy różniące poszczególne owady i tylko po jednym czarnym i żółtym pasku — bo te elementy są wspólne dla obu gatunków.

Aby teraz odzyskać nasz rój, należałoby pobrać z kopii zapasowej unikalne elementy każdej odzyskiwanej pszczoły, dodać do nich odpowiednią ilość elementów wspólnych (takich jak czarne i żółte paski) i pszczoła gotowa!

Zwróciłeś może uwagę na bardzo ciekawy wniosek wypływający z powyższego przykładu? Kiedy do urządzenia dokonującego deduplikacji danych przesyłane są różnego typu pliki, w pamięci masowej zapisywane są tylko unikalne bloki danych. Co więcej, gdy wartość współczynnika deduplikacji spada, to dzieje się tak tylko dlatego, że zwiększają się proporcje pomiędzy nowymi blokami a blokami już zapisanymi, a nie dlatego, że sam proces deduplikacji staje się mniej efektywny. W środowiskach dedykowanych do tworzenia kopii zapasowych ten efekt staje się bardzo wyraźny.

■ **Uwaga** Im więcej danych jest zapisanych w pamięci masowej urządzenia deduplikującego, tym mniejsza ilość danych jest zapisywana, ponieważ znacząco rośnie prawdopodobieństwo tego, że identyczny blok danych został już kiedyś wcześniej zapisany.

Mechanizm deduplikacji nie porównuje ze sobą bezpośrednio poszczególnych bloków danych, gdyż analizowanie każdego bloku danych i porównywanie go z już wcześniej zapisanymi byłoby operacją bardzo czasochłonną i pochłaniającą ogromne ilości zasobów systemu. Zamiast tego systemy deduplikujące używają specjalnego mechanizmu tworzenia

sygnatur, które pozwalają na jednoznaczną identyfikację poszczególnych bloków danych. Sygnatury bloków danych tworzone są poprzez przeprowadzenie odpowiednich obliczeń na zawartości tych bloków — taki proces jest znany pod nazwę **obliczania wartości funkcji skrótu** (ang. *hash calculation*). Obliczona wartość funkcji skrótu jest unikalnym identyfikatorem danego bloku danych oraz jego zawartości. Ponieważ wartości funkcji skrótu bloków danych mogą być sprawnie obliczane, łatwo ze sobą porównywane i szybko zapisywane w pamięci masowej, są wykorzystywane jako podstawa do identyfikacji bloków danych w systemach deduplikacji.

Wartość funkcji skrótu jest po prostu liczbą — zwykle dużą, zwykle unikatową, ale ciągle liczbą. Ponieważ ta liczba jest wyznaczana za pomocą standardowego, znanego algorytmu, bazującego na wartości poszczególnych bajtów bloku danych, teoretycznie jest możliwe wygenerowanie dwóch takich samych wartości funkcji skrótu dla dwóch różnych zbiorów danych. W takiej sytuacji mamy do czynienia z tzw. **kolizją funkcji skrótu** (ang. *hash collision*). Kiedy wystąpi kolizja funkcji skrótu, nie ma żadnej możliwości określenia, czy wyliczona wartość pochodziła z jednego, czy z drugiego bloku danych.

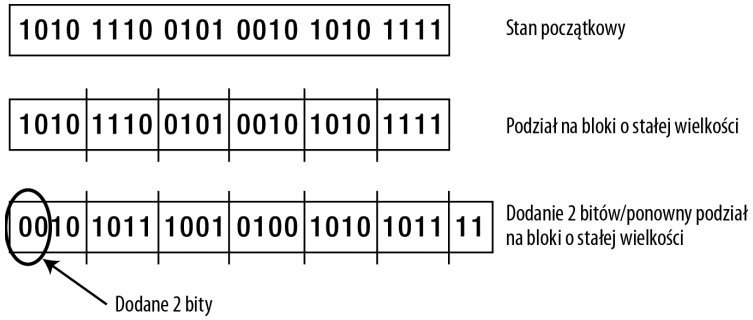
Jeżeli kolizja funkcji skrótu wystąpi w systemie deduplikującym, plik zawierający blok danych powodujący kolizję nie będzie mógł być poprawnie odtworzony. Dlaczego? Bloki danych, które generują kolizje funkcji skrótu, mają różne zawartości — jedna z nich oryginalnie należała do pliku, który chcemy odzyskać, ale druga pochodzi z zupełnie innego pliku. Jeżeli do odzyskiwania będzie użyty ten drugi blok danych, zawartość odzyskiwanego pliku zostanie uszkodzona i prawdopodobnie nie będziemy mogli z niego korzystać.

Na szczęście prawdopodobieństwo wystąpienia kolizji, choć teoretycznie możliwe, to jednak jest ekstremalnie niskie. Wartości funkcji skrótów w zależności od użytego algorytmu mają zwykle od 53 do 160 bitów długości (co daje rozpiętość wartości funkcji skrótu od 0 do 10^{48}). Nawet w tym najmniej korzystnym przypadku, jeżeli korzystamy z powszechnie znanego i używanego algorytmu obliczania funkcji skrótu dającego wynik o długości 53 bitów, prawdopodobieństwo wystąpienia kolizji wynosi jak 1 do 10^{-20}. Dla porównania możemy obliczyć, że (w zależności od przyjętej hipotezy) od początku znanego nam Wszechświata upłynęło do tej pory mniej niż 10^{17} sekund. Zatem jeżeli chciałbyś kolejno wygenerować wszystkie możliwe wartości funkcji skrótu, gdzie czas obliczania jednej wartości byłby poniżej sekundy, to i tak znalezienie wartości generującej kolizję zajęłoby więcej czasu, niż upłynęło do tej pory od Wielkiego Wybuchu.

Deduplikacja na poziomie bloków o stałej wielkości

W naszym przykładzie odnoszącym się do roju poszczególne elementy składowe pszczoły, lub inaczej poszczególne *bloki* składowe pszczoły, były takiej samej wielkości — możemy powiedzieć, że poszczególne pszczoły były dzielone zawsze dokładnie w taki sam sposób. Jeżeli chcemy zastosować deduplikację danych do plików w środowisku rzeczywistym, musimy mieć możliwość wyznaczenia odpowiedniego rozmiaru bloków danych w pliku. Istnieją dwa sposoby na osiągnięcie tego zamierzenia: możemy arbitralnie wybrać rozmiar bloku danych do deduplikacji lub określić rozmiar bloku na bazie wybranych znaczników w strumieniu danych.

Wiele systemów deduplikujących wykorzystuje mechanizm bazujący na stałej wielkości bloku danych. Jest to rozwiązanie relatywnie proste do wdrożenia, ponieważ wejściowy strumień danych jest po prostu dzielony na bloki o stałej wielkości, które są następnie analizowane, identyfikowane i porównywane z blokami zapisanymi już wcześniej. Jeżeli blok danych o danej sygnaturze już istnieje w pamięci masowej, zostaje pominięty i zamiast niego zapisany zostaje tylko wskaźnik do istniejącego bloku danych. Wadą takiego rozwiązania jest to, co się dzieje, kiedy dane w strumieniu wejściowym ulegają zmianie, co zostało zilustrowane na rysunku 5.3.



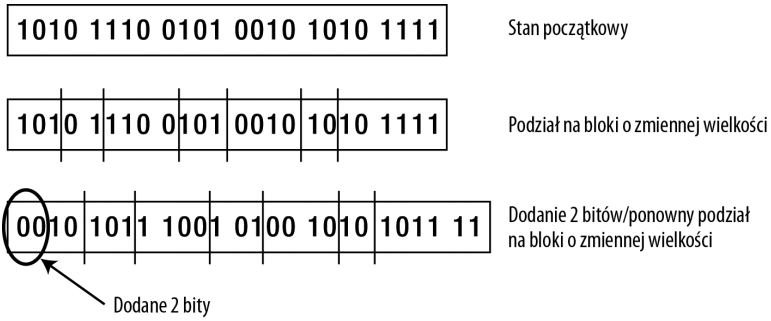
Rysunek 5.3. Deduplikacja na poziomie bloków o stałej wielkości

Pierwszy rysunek reprezentuje początkowy stan danych wejściowych, które zostają wstępnie zapisane jako prosta sekwencja bitów. Następnie dokonywany jest podział strumienia danych na bloki o stałej wielkości — w naszym przypadku każdy blok danych składa się z 4 bitów. Drugi rysunek ilustruje początkowy stan deduplikacji; jak widać w strumieniu danych zostały zidentyfikowane dwa identyczne bloki danych, które mogą zostać poddane procesowi deduplikacji. Następnie, na skutek innych operacji, na początku sekwencji danych dodawane są dwa bity, co powoduje przesunięcie miejsc podziału danych na bloki. Zauważ, co się teraz dzieje z deduplikacją. Pomimo iż nadal mamy dwa bloki danych mające identyczną zawartość, to jednak wszystkie kolejne bloki danych są inne niż przy poprzednim podziale na bloki. Co więcej, owe dwa identyczne bloki danych mają teraz zupełnie inną zawartość niż dwa bloki po pierwszym podziale, nie mówiąc już o tym, że po nowym podziale zawartość niemal wszystkich pozostałych bloków danych zmieniła się w stosunku do stanu początkowego. Taki rodzaj „nowych" danych ma z punktu widzenia deduplikacji wyjątkowo negatywny wpływ na ilość zapisywanych w pamięci masowej danych i znacząco redukuje wartość współczynnika deduplikacji, powodując niepotrzebne zapisywanie w pamięci masowej dodatkowych danych. Ilość „nowych" danych będących rezultatem przypadkowego przesunięcia okna podziału na bloki staje się znacząca zwłaszcza w przypadku bloków danych o większych rozmiarach. Większe rozmiary bloków danych mają większą liczbę kombinacji bitów wewnątrz bloku, a im większa liczba takich kombinacji, tym większe prawdopodobieństwo, że blok danych o takiej zawartości nie został jeszcze zapisany w pamięci masowej urządzenia deduplikującego.

Deduplikacja na poziomie bloku o zmiennej wielkości

Istnieje alternatywne rozwiązanie dla deduplikacji na poziomie bloku o stałej wielkości, które rozwiązuje wiele problemów powodujących zmniejszenie wydajności takiej deduplikacji: *deduplikacja* **na poziomie bloku o zmiennej wielkości**. W takim rozwiązaniu dokonywana jest analiza strumienia danych wejściowych i odszukiwane są odpowiednie „znaczniki", które wyznaczają punkty podziału bloków danych. Aby to zilustrować, powróćmy do naszego poprzedniego przykładu i zastosujmy do niego scenariusz z deduplikacją na poziomie bloków o zmiennej wielkości (patrz rysunek 5.4).

I znów analizę wejściowego strumienia danych rozpoczynamy od stanu początkowego i na bazie **znaczników** dokonujemy podziału na bloki. Wspomniane „znaczniki" są miarą prawdopodobieństwa zdarzenia polegającego na tym, że w analizowanym zbiorze danych pojawi się ponownie blok danych o zawartości wyznaczonej przez dany rozmiar bloku. W naszym przypadku rozmiary bloków wahają się od 2 do 6 bitów. Teraz do oryginalnego strumienia danych dodajemy dwa takie same bity jak w poprzednim przykładzie i dokonujemy



Rysunek 5.4. Deduplikacja na poziomie bloków o zmiennej wielkości

ponownego podziału na bloki. Dzięki odpowiedniemu mechanizmowi znaczników niemal wszystkie bloki danych mają swoje odpowiedniki w poprzednim, początkowym zestawie danych, dzięki czemu możemy tutaj osiągnąć przyzwoitą wartość współczynnika deduplikacji i znacząco zredukować ilość danych zapisywanych w pamięci masowej. Oczywiście jest to bardzo uproszczony przykład, ale dobrze ilustruje zasadę podziału na bloki danych o zmiennej wielkości. W rzeczywistości algorytmy wyznaczające miejsca podziału bloków danych są bardzo złożone i są zwykle opatentowane przez producentów poszczególnych rozwiązań.

Ograniczenia typów danych: multimedia, tajemnice i Matka Natura

Deduplikacja nie jest jednak uniwersalnym rozwiązaniem pozwalającym na efektywne tworzenie kopii zapasowych wszystkich rodzajów danych. Rodzaje danych, które z pewnością nie będą najlepszymi kandydatami do deduplikacji, można określić jako MTN: multimedia, tajemnice i natura.

Pliki multimedialne, takie jak *.mp3*, *.jpg* i wiele innych formatów, zazwyczaj zawierają skompresowane dane lub dane, w których z definicji nie ma zbyt wielu powtarzalnych wzorców. Dokonanie kompresji przed utworzeniem kopii zapasowej skutecznie eliminuje ze strumienia danych powtarzające się elementy (jak wiadomo, proces kompresji ma za zadanie usunięcie takich elementów i zastąpienie ich odpowiednimi znacznikami i wskaźnikami). Większość plików multimedialnych wykorzystuje taki czy inny rodzaj kompresji w celu zmniejszenia rozmiaru pliku przy zachowaniu jakości przechowywanego obrazu czy dźwięku. Nawet pliki zapisane w bezstratnych formatach multimedialnych nie poddają się zbyt dobrze procesom kompresji czy deduplikacji. Aby zilustrować przyczyny tego zjawiska, wyobraź sobie, z czego składa się zdjęcie cyfrowe — to zbiór odpowiednio zakodowanych danych, reprezentujących wartości kolorów składowych (czerwonego, zielonego i niebieskiego) dla poszczególnych pikseli. Każdy moment wykonywania zdjęcia jest w jakimś sensie unikalny i nigdy nie będzie można go dokładnie powtórzyć. Pomimo iż kolejne zdjęcia mogą mieć jakieś składniki wspólne, to jednak kompozycja kolejnych zdjęć będzie się od siebie różnić, nawet jeżeli poszczególne zdjęcia były wykonywane w szybkiej sekwencji. Z binarnego punktu widzenia prowadzi to do utworzenia wielu różniących się od siebie binarnych wzorców danych, które prawdopodobnie nigdy wcześniej nie były analizowane przez dany system deduplikujący, co z kolei prowadzi do konieczności ich zapisania i tym samym obniżenia współczynnika deduplikacji.

Określenie **natura** dotyczy danych, które są gromadzone i przetwarzane w wyniku doświadczeń i badań naukowych, czy zbierane z sensorów analizujących zjawiska natu-

ralne, takie jak dane hydrologiczne czy sejsmiczne. Na pierwszy rzut oka rozkład wartości takich danych może być w dużej mierze losowy, co z punktu widzenia deduplikacji nie jest zbyt dobrym znakiem — dane losowe nie poddają się zbyt dobrze deduplikacji.

Wreszcie **tajemnice**, czy inaczej mówiąc pliki zaszyfrowane, również nie są najlepszymi kandydatami do deduplikacji. Ideą szyfrowania danych jest ukrycie ważnych informacji za ścianą pozornie losowych danych. Z kolei losowość danych stanowi przekleństwo dla deduplikacji. W zasadzie zaszyfrowane dane powinny zostać przed deduplikacją odszyfrowane lub w przeciwnym razie współczynnik deduplikacji nie będzie zbyt imponujący i zwykle zbliżony do 1:1.

■ **Uwaga** Warto tutaj powiedzieć kilka słów na temat bezpieczeństwa danych poddawanych procesowi deduplikacji. Po zapisaniu danych poddanych procesowi deduplikacji możemy powiedzieć, że zostały one w pewnym stopniu zaszyfrowane, ponieważ podczas takiej operacji kopiowane pliki zostają rozbite na unikatowe bloki danych. Kolejne bloki danych mogą pochodzić z różnego typu plików binarnych, tekstowych, dokumentów i tak dalej i nie są powiązane z żadnym konkretnym typem plików. Dzięki temu poszczególne bloki nie zawierają danych pozwalających na jednoznaczne przypisanie ich do konkretnych plików bez dogłębnej znajomości technologii umożliwiającej zrekonstruowanie całego pliku z serii wskaźników i unikatowych bloków danych. Ponieważ wszystkie bloki danych są poindeksowane, a indeks nie ma żadnego powiązania z miejscem wystąpienia danego bloku w pliku źródłowym, dane zapisywane w pamięci masowej urządzenia deduplikującego mają charakter pseudolosowy, co jest całkiem przyzwoitym sposobem szyfrowania. Ponowne szyfrowanie bloków danych po deduplikacji jest zbędne, ponieważ prowadzi do szyfrowania danych, które już zostały w dosyć efektywny sposób „zaszyfrowane". Oczywiście można kwestionować bezpieczeństwo danych zapisanych w ten sposób, ale w praktyce szanse na zrekonstruowanie wybranego pliku po deduplikacji, gdy ma się do dyspozycji tylko zestaw wskaźników i unikatowych bloków danych i nie dysponuje się szczegółowym opisem kolejności bloków, są naprawdę minimalne.

Rodzaje i definicje deduplikacji

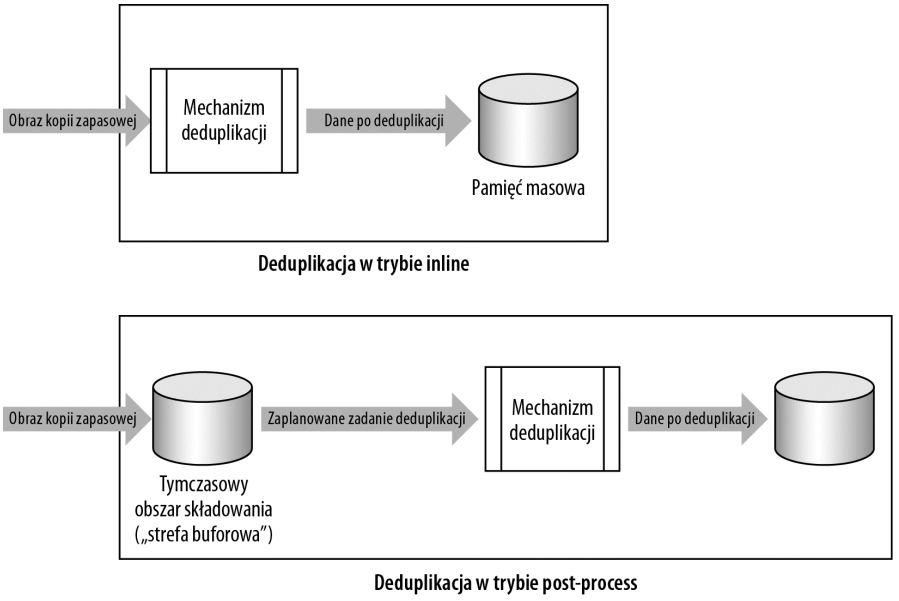
Omówiliśmy już teoretyczne zasady działania deduplikacji, ale w jaki sposób takie rozwiązanie jest wdrażane w praktyce? Zazwyczaj system deduplikujący ma postać oprogramowania osadzonego w dedykowanym urządzeniu lub zainstalowanego jako część systemu tworzenia kopii zapasowych. Oprogramowanie deduplikujące może być zaimplementowane na jeden z dwóch sposobów:

- **Deduplikacja po stronie źródła** (ang. *Source-based deduplication*) — w tym scenariuszu deduplikacja jest przeprowadzana po stronie klienta i do systemu pamięci masowej przesyłany jest wynikowy strumień danych.
- **Deduplikacja po stronie celu** (ang. *Target-based deduplication*) — deduplikacja jest przeprowadzana przez urządzenie zapisujące dane w pamięci masowej po przesłaniu przez klienta pełnego strumienia danych kopii zapasowej.

Oba rodzaje deduplikacji mają swoje zalety i ograniczenia uzależnione od rodzaju kopiowanych danych.

W modelu deduplikacji przeprowadzanej po stronie systemu docelowego możliwe są dwa tryby przetwarzania danych:

- W trybie *inline* deduplikacja danych jest wykonywana „w locie", czyli wejściowy strumień danych jest przetwarzany w czasie rzeczywistym i od razu zapisywany w pamięci masowej.
- W trybie *post-process* napływające dane są najpierw zapisywane w tymczasowym obszarze składowania (strefie buforowej) w tradycyjny sposób, a dopiero gdy ten proces się zakończy, są ponownie przeglądane i poddawane deduplikacji (patrz rysunek 5.5).



Rysunek 5.5. Porównanie deduplikacji w trybie inline oraz w trybie post-process

Zaletą **deduplikacji w trybie inline** jest to, że wymaga tylko przestrzeni pamięci masowej, która jest niezbędna do zapisywania danych po zakończeniu procesu deduplikacji. Dzięki temu rozmiar wymaganej pamięci masowej może być zredukowany do niezbędnego minimum. Z drugiej jednak strony, maksymalna przepustowość urządzenia pracującego w takim trybie może być mniejsza niż w przypadku urządzeń pracujących w trybie post-process, ponieważ deduplikacja musi być przeprowadzona jeszcze przed zapisaniem danych w pamięci masowej. W zależności od architektury urządzenia proces zapisu w pamięci masowej lub odzyskiwania danych może być wolniejszy, ponieważ system będzie potrzebował dodatkowego czasu niezbędnego do analizy strumienia danych pod kątem powtarzających się bloków danych i odpowiednio zapisywania ich w pamięci masowej lub rekonstrukcji odzyskiwanego pliku. Deduplikacja w trybie inline jest najczęściej wykorzystywana w urządzeniach dedykowanych, takich jak EMC DataDomain, aczkolwiek jest również stosowana w rozwiązaniach NetBackup PureDisk oraz CommVault.

Z drugiej strony, **deduplikacja w trybie post-process** wykorzystuje pewną część pamięci masowej jako tymczasowy obszar składowania dla strumienia danych kopii zapasowej. Takie podejście pozwala na zapisywanie strumienia danych z maksymalną szybkością, ale wymaga też zapewnienia dodatkowego obszaru pamięci masowej o znacznych rozmiarach — tymczasowy obszar składowania musi być na tyle duży, aby pomieścić cały strumień danych przesyłany z klienta. Dopiero po zakończeniu zapisywania danych następuje proces deduplikacji i przetworzone dane są przenoszone do obszaru docelowego. Deduplikacja

w trybie post-process jest spotykana zarówno w rozwiązaniach dedykowanych, takich jak seria urządzeń Quantum Dxi, jak i w rozwiązaniach pomocniczych umożliwiających tworzenie dodatkowych obrazów kopii zapasowych z systemów głównych na nośnikach dodatkowych, takich jak proces Vault w aplikacji NetBackup czy proces AUX dla CommVault.

Deduplikacja po stronie źródła

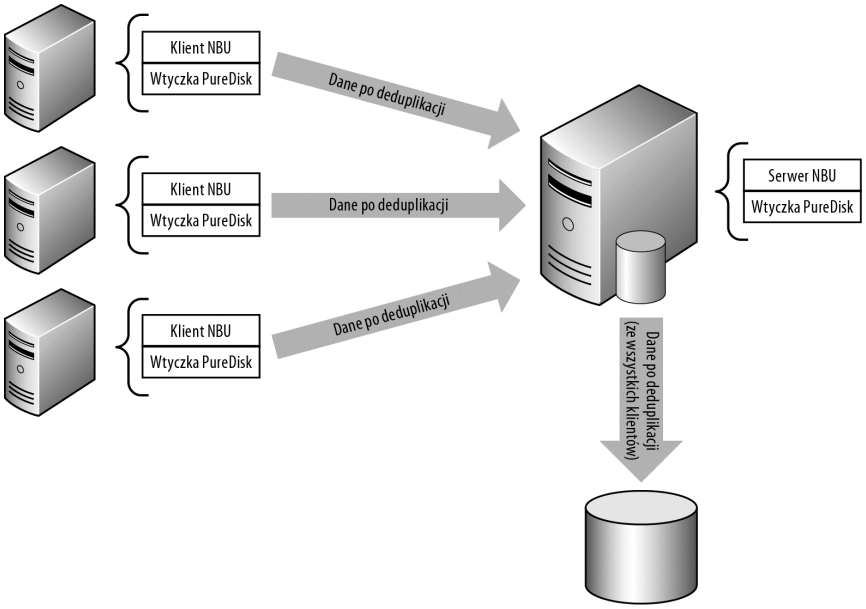
Jednym z rodzajów deduplikacji jest deduplikacja po stronie źródła. W takim scenariuszu dane są deduplikowane *przed* wysłaniem przez sieć IP do zapisania w pamięci masowej. Poprzez wykonanie deduplikacji przez klienta można znacząco zredukować ilość danych przesyłanych przez sieć, ponieważ przesyłane są tylko unikatowe bloki danych — wszystkie inne, powtarzające się bloki danych zostały wcześniej wyeliminowane przez proces deduplikacji. Na rynku możemy spotkać dwa główne produkty oferujące ten rodzaj deduplikacji: NetBackup oraz EMC Avamar. Chociaż w naszej książce koncentrujemy się głównie na pakietach CommVault i NetBackup, warto zauważyć, że rozwiązania firmy CommVault, takie jak Simpana 8, wykorzystują proces deduplikacji po stronie celu, czyli przeprowadzają deduplikację dopiero po zapisaniu całego strumienia danych w tymczasowym obszarze składowania. Z pewnością warto sprawdzić, jak sprawuje się to rozwiązanie w porównaniu z Avamarem. W środowisku deduplikacji po stronie źródła klient przed wysłaniem danych na serwer wykorzystuje specjalny mechanizm do określenia, które bloki danych są unikatowe.

Aby wykonać deduplikację danych po stronie źródła, standardowe klienty aplikacji NetBackup wyposażone są w specjalne wtyczki PureDisk (ang. *PureDisk plug-ins*), dzięki którym możliwa staje się lokalna deduplikacja przed wysłaniem strumienia danych na serwer kopii zapasowych poprzez sieć TCP/IP. Urządzeniem końcowym może być dedykowane urządzenie NetBackup lub inne rozwiązanie oparte na standardowym serwerze typu MS (ang. *Media Server*). Klient aplikacji NetBackup wysyła tylko unikatowe bloki danych i pozostawia ich dalsze przetwarzanie (dokończenie procesu deduplikacji) serwerowi kopii zapasowych (patrz rysunek 5.6).

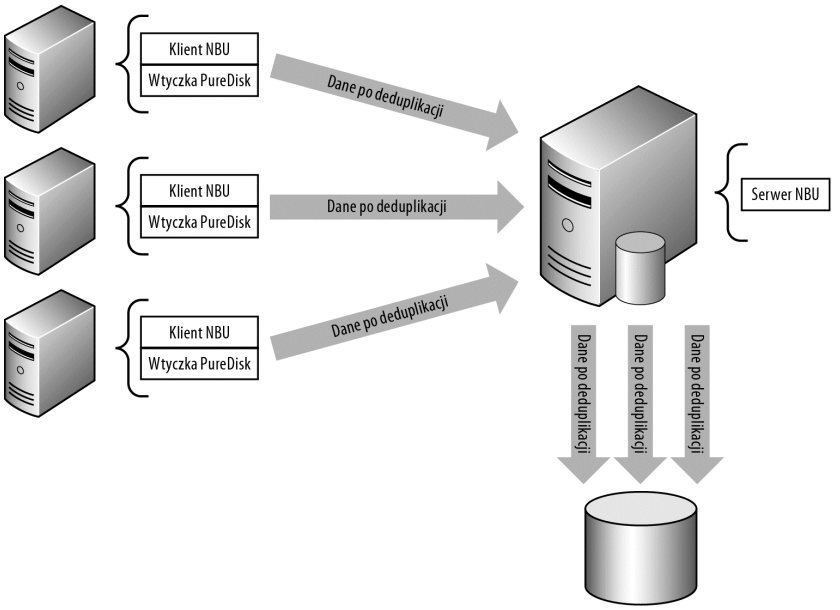
Opisane powyżej rozwiązanie zapewnia przeprowadzanie deduplikacji na poziomie poszczególnych klientów i dzięki temu pozwala na znaczące zredukowanie ilości danych przesyłanych po sieci na serwer. Jednak ze względu na fakt, że w takim modelu nie ma deduplikacji danych pomiędzy klientami, istnieje potencjalna możliwość, że taki sam pakiet danych zostanie przesłany na serwer z dwóch lub więcej różnych klientów. Jeśli jednak sam proces deduplikacji na serwerach działa poprawnie i rozmiar bloku danych jest względnie mały, to nie powinno to mieć większego wpływu na obciążenie sieci.

Ponieważ klient deduplikacji pakietu NetBackup może korzystać ze zwykłego serwera MS, do przechowywania przetworzonych danych można wykorzystać standardowy serwer NetBackup. W takim rozwiązaniu podstawowa architektura strefy danych NetBackup nie ulega zmianie, aczkolwiek po stronie pamięci masowej implementacja infrastruktury serwera MS może być znacznie bardziej złożona (patrz rysunek 5.7). Więcej informacji na ten temat przedstawimy niebawem, podczas omawiania deduplikacji po stronie celu.

Dla porównania: rozwiązanie EMC Avamar wykorzystuje zupełnie inny model. Co prawda nadal korzysta ono z deduplikacji danych po stronie klienta, ale oprócz tego zapewnia deduplikację danych na poziomie globalnym. Zadanie to jest realizowane poprzez złożony proces, który generuje funkcje skrótu dla bloków danych na poziomie poszczególnych klientów, a następnie przed wysłaniem na serwer porównuje z centralnym repozytorium. Takie rozwiązanie zapewnia wiele zalet. Po pierwsze, następuje dalsza redukcja ilości danych przesyłanych na serwer, ponieważ przesyłane są tylko bloki danych, które są unikatowe dla wszystkich klientów. Po drugie, kiedy do systemu podłączymy kolejnego klienta, to na serwer będą wysyłane z niego tylko takie bloki danych, które nie mają swoich odpowiedników w centralnej pamięci masowej. Z drugiej strony, pamięć masowa takiego rozwiązania jest dedykowanym urządzeniem, którego pojemność można zwiększać poprzez



Rysunek 5.6. Deduplikacja danych na klientach NetBackup z wysyłaniem danych na serwer MS z dodatkową deduplikacją



Rysunek 5.7. Deduplikacja danych na klientach NetBackup z wysyłaniem danych na serwer MS bez dodatkowej deduplikacji

dodawanie kolejnych węzłów. Ponieważ dodawanie nowych węzłów „pożera" całkiem spory obszar przestrzeni fizycznej, należy się starać, aby rozwiązanie EMC Avamar było stosowane tylko do przechowywania kopii zapasowych danych, które dobrze poddają się procesowi deduplikacji, co pozwoli utrzymywać liczbę węzłów niezbędnych do poprawnego funkcjonowania całego systemu.

Deduplikacja po stronie celu

W przeciwieństwie do systemów omawianych w poprzednim punkcie w modelu **deduplikacji po stronie celu** sam proces deduplikacji jest wykonywany na urządzeniu docelowym dopiero po całkowitym zakończeniu zapisywania danych przesyłanych przez klientów. NetBackup oferuje kilka różnych rozwiązań wdrożenia deduplikacji po stronie celu, są to między innymi NetBackup Media Server with Deduplication, PureDisk oraz NetBackup PureDisk Option. Pakiet CommVault Simpana posiada tylko jedno rozwiązanie oparte na serwerze Media Agent, które zapewnia deduplikację danych po stronie celu dla struktury CommCell. Oprócz tych rozwiązań istnieją również inne, oparte na dedykowanych urządzeniach do tworzenia kopii zapasowych, które są w pełni obsługiwane zarówno przez pakiet NetBackup, jak i CommVault. Urządzenia takie pozwalają na przejęcie procesu deduplikacji od serwerów MS/MA.

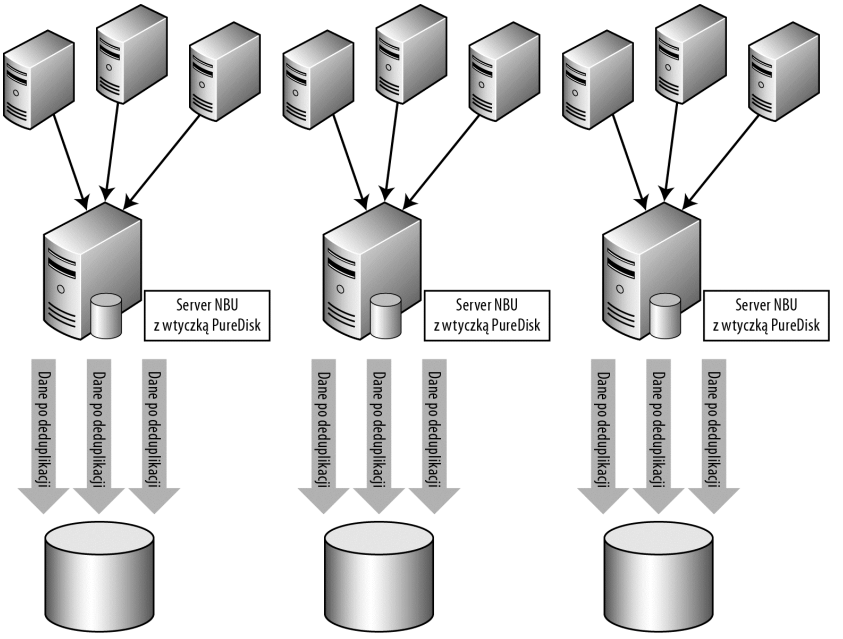
Nasuwa się zatem pytanie: jeżeli zastosowanie rozwiązań deduplikacji po stronie źródła pozwala na znaczące zredukowanie ilości danych do przetwarzania jeszcze przed ich wysłaniem od klienta, dlaczego w ogóle stosować deduplikację po stronie celu? Istnieje co najmniej kilka powodów. Po pierwsze, deduplikacja po stronie źródła nie załatwia wszystkiego. Istnieją praktyczne ograniczenia ilości danych, które mogą zostać przetworzone w jednostce czasu nawet przy użyciu bardzo efektywnych algorytmów obliczania funkcji skrótu. Spowodowane jest to ograniczoną wydajnością lokalnych jednostek CPU, wydajnością docelowej pamięci masowej oraz — w przypadku rozwiązania EMC Avamar — koniecznością ciągłego komunikowania się z centralnym serwerem w celu sprawdzenia przed wysłaniem, czy poszczególne bloki danych są unikatowe. Systemy, w których duża ilość danych ulega częstym zmianom, generują w efekcie duże wolumeny danych, które muszą zostać przetworzone, i zwykle odznaczają się większą ilością unikatowych bloków danych, które muszą zostać wysłane na urządzenie docelowe. Deduplikacja po stronie źródła jest rozwiązaniem, które najlepiej sprawdza się dla źródeł danych o niskim współczynniku zmian oraz systemów, w których znajduje się ogromna liczba małych plików, znanych pod nazwą systemów plików o dużej gęstości (HDFS — ang. *High Density File System*).

Dla innych zestawów danych, a zwłaszcza dla dużych baz danych, ilość zasobów systemowych niezbędnych do skutecznego przeprowadzenia procesu deduplikacji wymaga zastosowania specjalnego, dedykowanego rozwiązania — stąd potrzeba wprowadzenia scenariusza deduplikacji po stronie celu.

NetBackup

Dla takiego scenariusza NetBackup zapewnia dwa podstawowe rozwiązania w dwóch wariantach. Najbardziej podstawowym rozwiązaniem jest implementacja deduplikacji na serwerze NetBackup Media Server i zastosowanie do przechowywania przetworzonych danych dedykowanej jednostki DSU (ang. *Disk Storage Unit*). Ogólna architektura takiego rozwiązania nie różni się zbytnio od standardowego wdrożenia innych rozwiązań NetBackup, ale jednak istnieją pewne różnice. W modelu wykorzystującym serwer MS klienci wykonujący deduplikację danych, serwer MS zapisujący dane w pamięci masowej oraz dowolne inne serwery pomocnicze są grupowane w jednostki nazywane węzłami. **Węzeł deduplikacji** wyznacza zasięg logicznej hermetyzacji granic obszaru, w którym zachodzi proces deduplikacji danych — klienci należący do innego węzła oraz ich dane nie są w żaden sposób deduplikowane ani zapisywane względem danych bieżącego węzła.

Deduplikacja wielosystemowa, taka jak w opisanym wyżej modelu z serwerem MS, pozwala na ograniczenie zasięgu deduplikacji tylko do klientów powiązanych z danym serwerem MS (patrz rysunek 5.8).

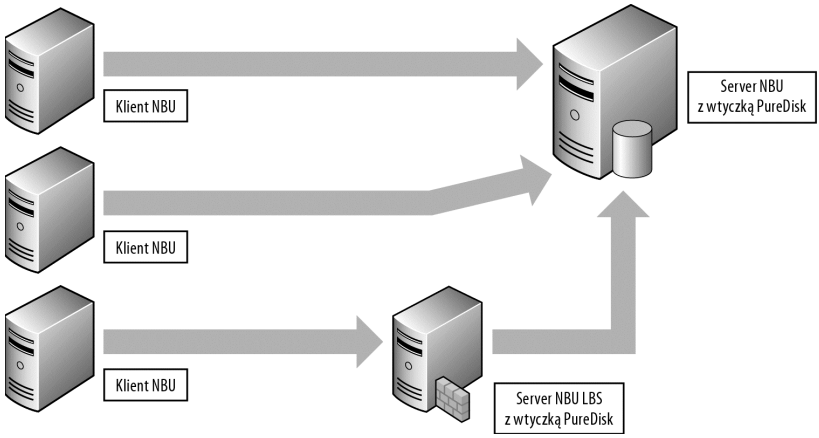


Rysunek 5.8. Ograniczanie zasięgu deduplikacji na serwerach NetBackup Media Servers

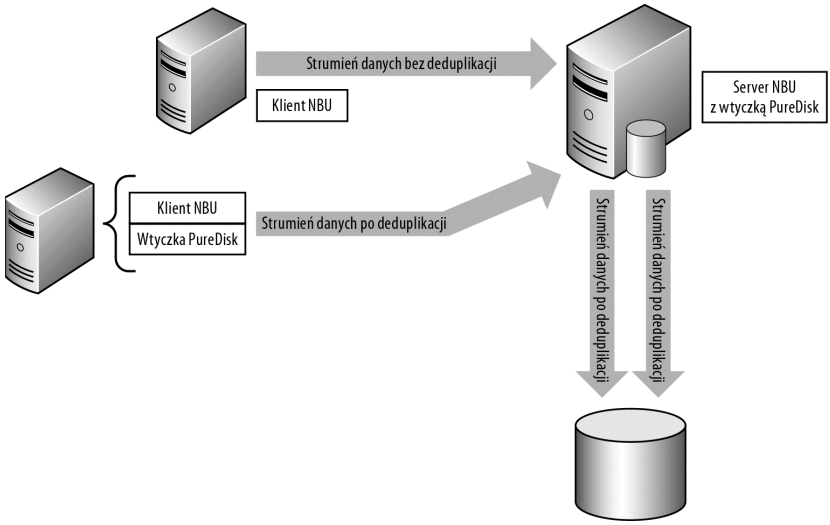
W typowych rozwiązaniach serwery MS zapewniają całość przetwarzania danych, włączając w to obliczanie wartości sygnatur poszczególnych bloków danych (zwanych czasem „odciskami palca" bloku danych), zarządzanie pamięcią masową, w której przechowywane są unikatowe bloki danych, oraz przesyłanie metadanych do głównego serwera zarządzającego całym procesem. Oprócz tego w każdym z węzłów można umieścić dodatkowy serwer pomocniczy *LBS* (ang. *Load Balancing Server*), którego zadaniem jest równomierne rozkładanie obciążenia poszczególnych serwerów węzła. Server LBS pozwala na zmniejszenie obciążenia głównego serwera deduplikującego obliczeniami sygnatur bloków danych, a co za tym idzie, na zwiększenie wydajności całego rozwiązania (patrz rysunek 5.9).

Z drugiej strony, instalowanie dodatkowego serwera LSB nie jest zalecane dopóty, dopóki zasoby głównego serwera deduplikującego nie są wyczerpane i serwer nie jest w stanie przetwarzać już większej ilości danych w oknie tworzenia kopii zapasowej. Jest to spowodowane tym, że serwer LSB wprowadza jednak pewne opóźnienia do procesu tworzenia kopii zapasowych i dostarcza tylko usługi obliczeniowe. Serwer LSB nie ma żadnego wpływu na zarządzanie i zapis danych w pamięci masowej.

Z kolei zaletą takiego rozwiązania jest to, że w obrębie danego węzła można używać zarówno klientów deduplikujących dane, jak i klientów pozbawionych tej możliwości. Dzięki temu deduplikacja danych po stronie źródła ma miejsce na przystosowanych do tego celu klientach, a dane przesyłane przez pozostałe klienty będą deduplikowane po stronie celu. Strumienie danych kopii zapasowych obu rodzajów klientów mogą być łączone i poddawane ostatecznej deduplikacji na głównym serwerze MS, co pozwala na utworzenie uniwersalnego systemu łączącego deduplikację po stronie źródła i po stronie celu w jeden wydajny system tworzenia kopii zapasowych (patrz rysunek 5.10).



Rysunek 5.9. Zastosowanie serwera NetBackup LBS



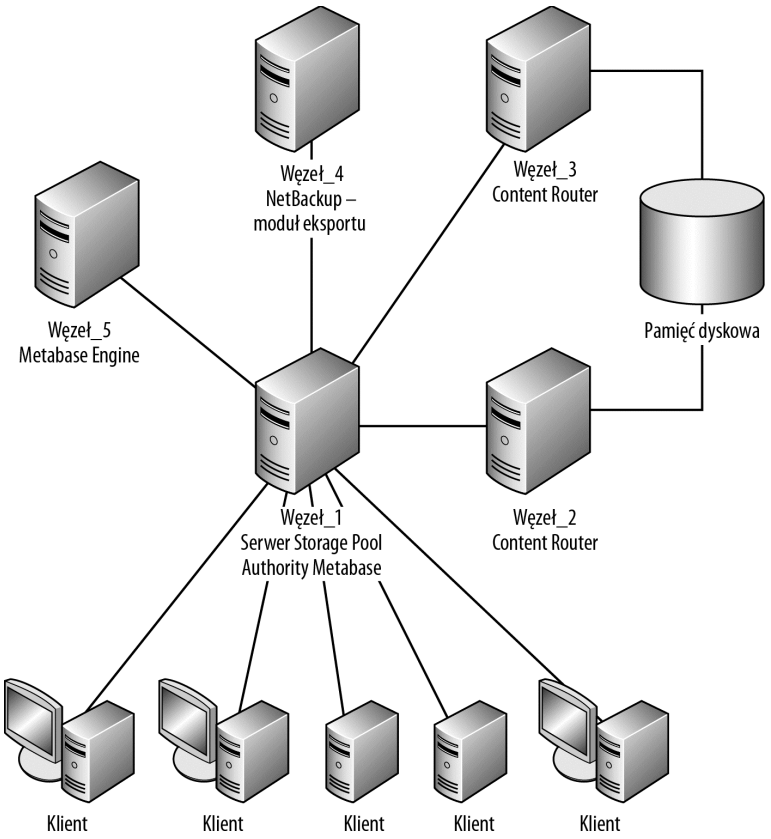
Rysunek 5.10. Uniwersalny system NetBackup Media Server z deduplikacją po stronie źródła i po stronie celu

Kolejnym popularnym rozwiązaniem jest zastosowanie samodzielnego systemu PureDisk. Podczas gdy wtyczka PureDisk opisywana poprzednio udostępnia tylko podstawowe mechanizmy deduplikacji, pełny system PureDisk jest samodzielnym produktem (w zasadzie tylko do pewnego stopnia, ale o tym za chwilę), który udostępnia możliwość tworzenia deduplikowanych kopii zapasowych dla wielu klientów. Rozwiązanie PureDisk posiada unikalną architekturę, która jest zupełnie inna niż opisywany wcześniej węzeł deduplikacji oparty na serwerze MS. PureDisk Storage Pool to kolekcja serwerów i klientów, która zapewnia skalowalną metodę deduplikacji danych przy użyciu urządzeń bazujących na standardowych rozwiązaniach sprzętowych. PureDisk został utworzony w oparciu o system PDOS (ang. *PureDisk Operating System*), który działa na sprzęcie o standardowej architekturze Intel x64. Dzięki temu PureDisk posiada wiele zalet urządzeń dedykowanych bez konieczności ponoszenia dodatkowych wydatków.

Aby można było skorzystać z technologii PureDisk, trzeba w systemie zainstalować cały szereg dodatkowych usług, takich jak:

- *Storage Pool Authority* — menedżer zarządzania pulą pamięci masowej.
- *Metabase Engine* — baza danych przechowująca wybrane metadane opisujące klientów oraz ich kopie zapasowe.
- *Metabase Server* — usługa zarządzająca zapytaniami do serwera Metabase Engine.
- *Content Router* — usługa zarządzania obszarem przechowywania danych po deduplikacji.

Poszczególne usługi muszą być zainstalowane na co najmniej dwóch serwerach, tak aby uzyskać odpowiednią wydajność pamięci masowej. Usługi Metabase Engine oraz Content Router to dwie usługi dodawane do puli pamięci masowej, pozwalające na jej rozbudowę (patrz rysunek 5.11).

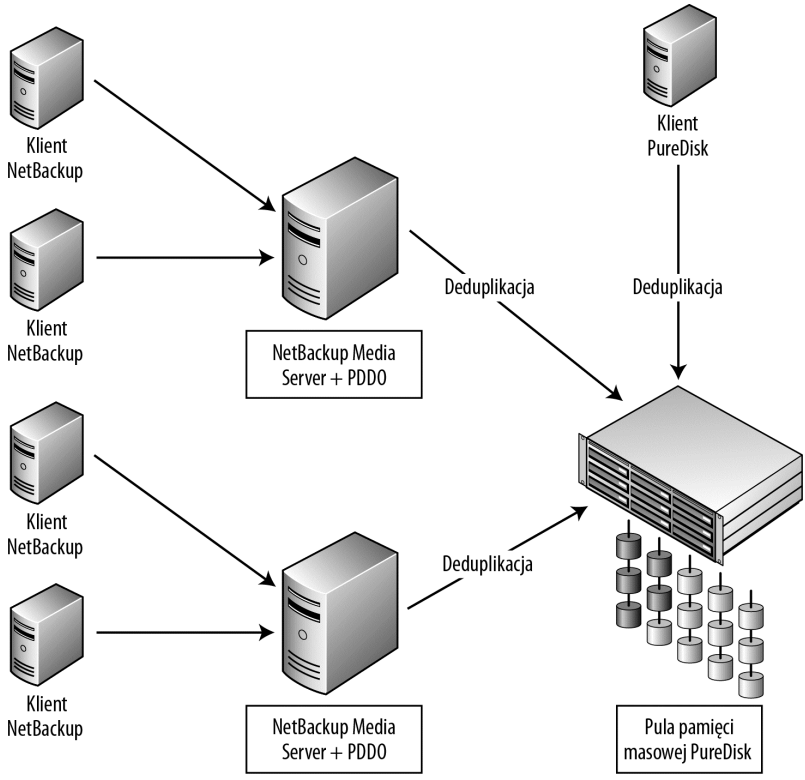


Rysunek 5.11. Rozwiązanie PureDisk[1]

[1] Według podręcznika *PureDisk: Getting Started Guide* — Symantec Corp., str. 18.

Każda z tych usług jest zazwyczaj instalowana na osobnym serwerze fizycznym w celu zapewnienia odpowiedniej wydajności operacji wejścia-wyjścia oraz przetwarzania danych przesyłanych do puli pamięci masowej (ang. *Storage Pool*), jak nazywana jest ta kolekcja usług. Po utworzeniu puli Storage Pool na poszczególnych klientach instalowane jest oprogramowanie PureDisk. Jeżeli klient będzie korzystał zarówno ze standardowego środowiska NetBackup, jak i rozwiązania PureDisk, oprogramowanie klienta NetBackup zostanie uzupełnione wtyczką PureDisk. Po zainstalowaniu oprogramowania klient jest podłączany do puli Storage Pool i tworzone są dla niego odpowiednie reguły tworzenia kopii zapasowych.

Choć obie usługi mogą działać niezależnie — i bardzo często tak właśnie się dzieje — NetBackup posiada inne opcje, pozwalające na połączenie tych dwóch metod deduplikacji po stronie celu w osobne rozwiązanie, dostosowane do zmian i wymagań danego środowiska. Jedną z bardziej interesujących kombinacji jest zdolność do połączenia standardowego serwera NetBackup Media Server z pulą PureDisk Storage Pool za pośrednictwem wtyczki NetBackup PureDisk Option. Pozwala to standardowym klientom NetBackup na tworzenie kopii zapasowych na standardowym serwerze MS, ale z możliwością skorzystania z zalet, jakie daje deduplikacja po stronie celu (patrz rysunek 5.12).



Rysunek 5.12. Zastosowanie standardowych serwerów MS w połączeniu z rozwiązaniem PureDisk[2]

Druga konfiguracja pozwala na rozwiązanie odwrotne. Opcja NetBackup PureDisk Connector pozwala standardowym klientom PureDisk na tworzenie swoich kopii zapasowych w standardowym środowisku NetBackup. Jest to realizowane poprzez instalację

[2] Według podręcznika *PureDisk Option Guide* — Symantec Corp., str. 14.

modułu eksportu NetBackup w puli PureDisk Storage Pool. Moduł eksportu pozwala na zrekonstruowanie plików i umieszczenie ich na tradycyjnych nośnikach wykorzystywanych przez NetBackup, włączając w to nośniki taśmowe, oraz zarządzanie plikami jak na tradycyjnym serwerze NetBackup Master. Dzięki temu możliwe staje się ukierunkowane odzyskiwanie danych, odzyskiwanie danych bezpośrednio na inne klienty niż te, z których pochodziła dana kopia zapasowa, oraz odzyskiwanie danych z klientów PureDisk na klienty NetBackup.

CommVault

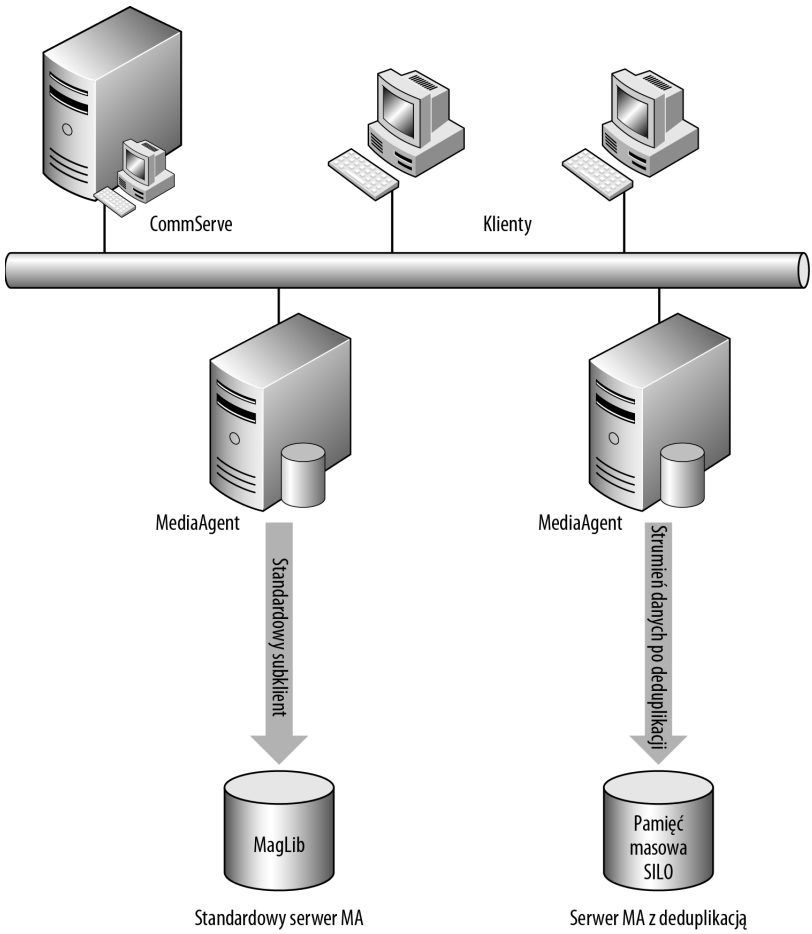
NetBackup posiada cztery sposoby realizacji deduplikacji po stronie celu — Media Server with deduplication Storage Pool, NetBackup PureDisk, opcję NetBackup PureDisk dla standardowych serwerów MS oraz NetBackup PureDisk Connector. Pakiet CommVault korzysta tylko z jednego rozwiązania — deduplikacji po stronie celu na serwerach Media Agent. Kiedy serwer MA jest konfigurowany pod kątem deduplikacji, tworzony na nim jest specjalny magazyn Deduplication Store. Jest to w zasadzie baza danych zawierająca sygnatury bloków danych oraz informacje o ich położeniu w pamięci masowej MagLib, w której przechowywane są dane po deduplikacji.

Deduplikacja odbywa się na poziomie subklienta, dzięki czemu może być selektywnie wdrożona dla całego klienta lub tylko wybranej części jego zasobów. Sygnatury bloków danych, dla których ma zostać utworzona kopia zapasowa, mogą być generowane zarówno po stronie klienta, jak i na serwerze MediaAgent. Pozwala to na elastyczność w przydzielaniu tego zadania dla jednej lub drugiej platformy, w zależności od ich bieżącego obciążenia. Implementacja deduplikacji po stronie celu na serwerze MediaAgent nie wymaga instalowania żadnych dodatkowych klientów. Ogólna architektura całego rozwiązania CommCell również się nie zmienia, dzięki czemu deduplikacja może być zaimplementowana po prostu jako nowy rodzaj kopii zapasowych, a nie jako duża zmiana architektury całego środowiska (patrz rysunek 5.13).

Istnieją jednak pewne ograniczenia. Aby można było mieć kontrolę nad zajętością pamięci masowej MagLib, niezbędne jest okresowe usuwanie zawartości magazynu Deduplication Store i migrowanie danych na nośniki taśmowe lub do pamięci masowej SILO (jak jest nazywana standardowa pamięć masowa serwera w terminologii CommVault). Rozwiązanie takie powoduje przeniesienie danych z pamięci MagLib na inne nośniki, ale przy okazji powoduje anulowanie procesu deduplikacji, wymuszając rekonstrukcję bloków zapisanych wcześniej w magazynie Deduplication Store. Na szczęście CommVault pozwala również na deduplikację danych, które są przenoszone na inne rodzaje nośników.

Ciągła ochrona danych i replikacja zdalna

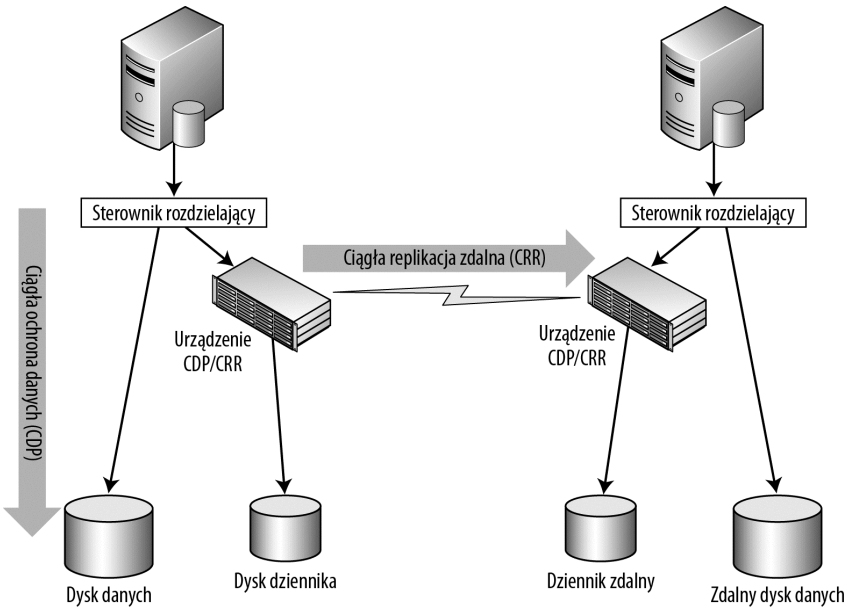
Zamiast tworzenia kopii zapasowych w regularnych odstępach czasu, co daje tylko szereg obrazów danych zapisanych w poszczególnych punktach czasu, bardzo interesująca byłaby możliwość posiadania kopii zapasowej danych z dowolnego punktu w czasie, możliwość odzyskiwania danych w okresie retencji do wybranego punktu w czasie z dokładnością na przykład do jednej minuty, replikowanie kopii zapasowej od razu podczas jej tworzenia i wykonywanie takich samych operacji w zdalnej siedzibie firmy. Takie wydawałoby się nieprawdopodobne wymagania to nic innego, jak tylko opis podstawowych możliwości nowej metody tworzenia kopii zapasowych, nazywanej ciągłą ochroną danych (CDP — ang. *Continuous Data Protection*) oraz ciągłą replikacją zdalną (CRR — ang. *Continuous Remote Replication*). CDP jest wykorzystywane do ochrony danych lokalnych (znajdujących się lokalnie na chronionym systemie), a zadaniem CRR jest replikacja kopii zapasowych na zdalne systemy pamięci masowych.



Rysunek 5.13. Porównanie rodzajów kopii zapasowych na serwerach MediaAgent

Jak to działa? Mechanizm CDP przechwytuje każdą operację wejścia-wyjścia na poziomie dysku i rozdziela ją na dwie operacje równoległe. Taki podział operacji wejścia-wyjścia jest możliwy dzięki zastosowaniu specjalnego sterownika, nazywanego **sterownikiem rozdzielającym** (ang. *splitter driver*) albo w skrócie **spliterem**. Dzięki spliterowi pierwsza operacja jest zapisywana na dysku tak jak zawsze, bez żadnych modyfikacji, a druga operacja jest zapisywana w specjalnym miejscu pamięci masowej, łącznie z numerem sekwencji przypisanym do tej operacji wejścia-wyjścia.

Miejsce na dysku, w którym rejestrowana jest historia zapisywanych bloków danych, nazywane jest **dziennikiem** (ang. *journal*). Oprogramowanie CDP tworzy osobny dziennik dla każdej zarządzanej jednostki LUN, stąd dostawcą pamięci masowej dla serwerów korzystających z rozwiązania CDP musi być SAN (ang. *Storage Area Network*). Oprócz numeru sekwencji operacji wejścia-wyjścia w dzienniku zapisywany jest dla każdego bloku danych również znacznik czasowy operacji, z dokładnością do milisekund. Oprogramowanie CDP może być zintegrowane z innymi aplikacjami, zazwyczaj są to aplikacje bazodanowe. Takie rozwiązanie pozwala na powiązanie bloków danych zarejestrowanych w dzienniku z poszczególnymi stanami aplikacji (patrz rysunek 5.14).

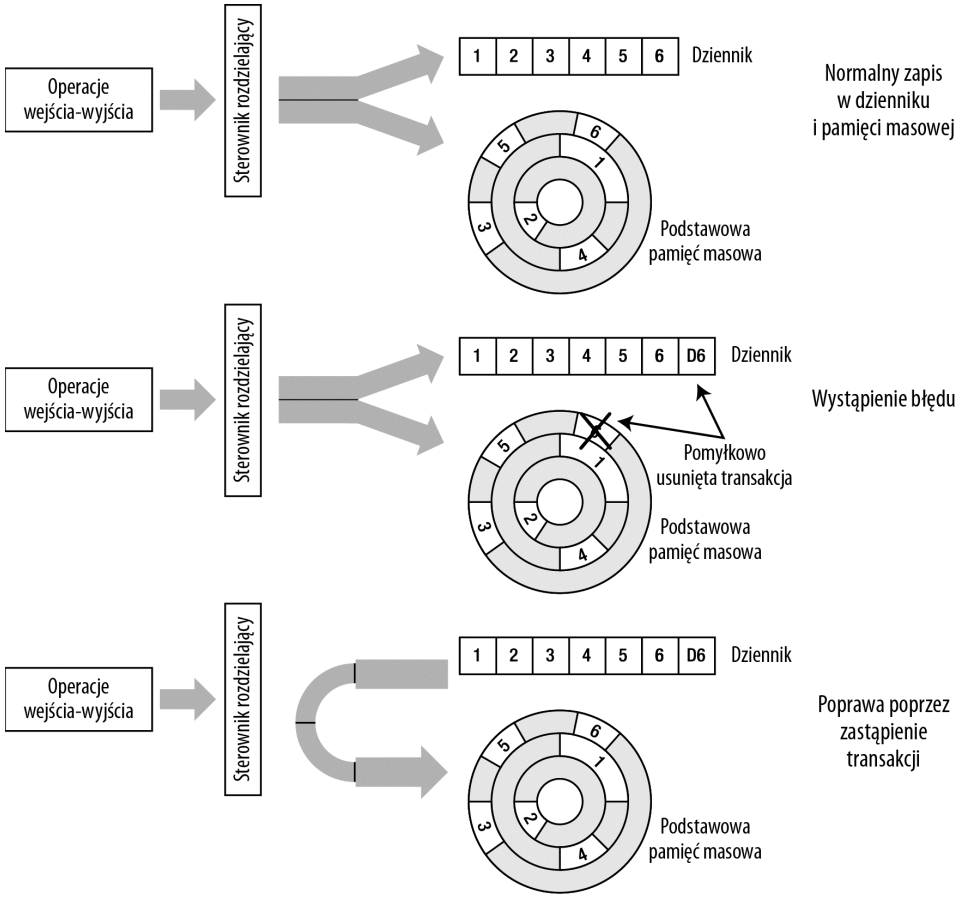


Rysunek 5.14. Technologie CDP i CRR

Takie rozwiązanie pozwala na efektywne tworzenie pełnostanowych kopii zapasowych danych dla dowolnego punktu w czasie. Jak? Ponieważ CDP przechowuje sekwencyjny listing wszystkich bloków danych powiązanych z chronionymi LUN-ami, zmiany w takich blokach mogą zostać zaaplikowane w odwrotnej kolejności, co pozwala na wycofywanie ostatnio wykonanych operacji aż do osiągnięcia określonego punktu w czasie. Poprzez powiązanie tego rozwiązania z aplikacją możemy zapewnić kopię zapasową stanu aplikacji z dowolnego punktu w czasie (mieszczącego się w okresie retencji) z dokładnością do pojedynczych milisekund lub do wybranego punktu kontrolnego (punktu przywracania) utworzonego przez aplikację. Poprzez sekwencyjne wycofywanie kolejnych transakcji można przywrócić dane do stanu w dowolnie wybranym momencie, dzięki czemu metoda CDP znakomicie sprawdza się jako wyrafinowany sposób tworzenia kopii zapasowych (patrz rysunek 5.15).

Dodatkowo większość rozwiązań CDP zapewnia możliwość publikacji dziennika (z poprzedniej chwili) łącznie z oryginalną zawartością pamięci masowej jako „wirtualnej" jednostki LUN, reprezentującej stan danych z określonego punktu w czasie, bez konieczności wycofywania poszczególnych transakcji z aktywnego LUN-a. Zazwyczaj taki wirtualny LUN pozwala na odczyt i zapis danych z opcją przeniesienia dokonanych na nim zmian do oryginalnego LUN-a i nadpisania jego stanu oraz z możliwością anulowania wszystkich dokonanych zmian po zwolnieniu wirtualnego LUN-a.

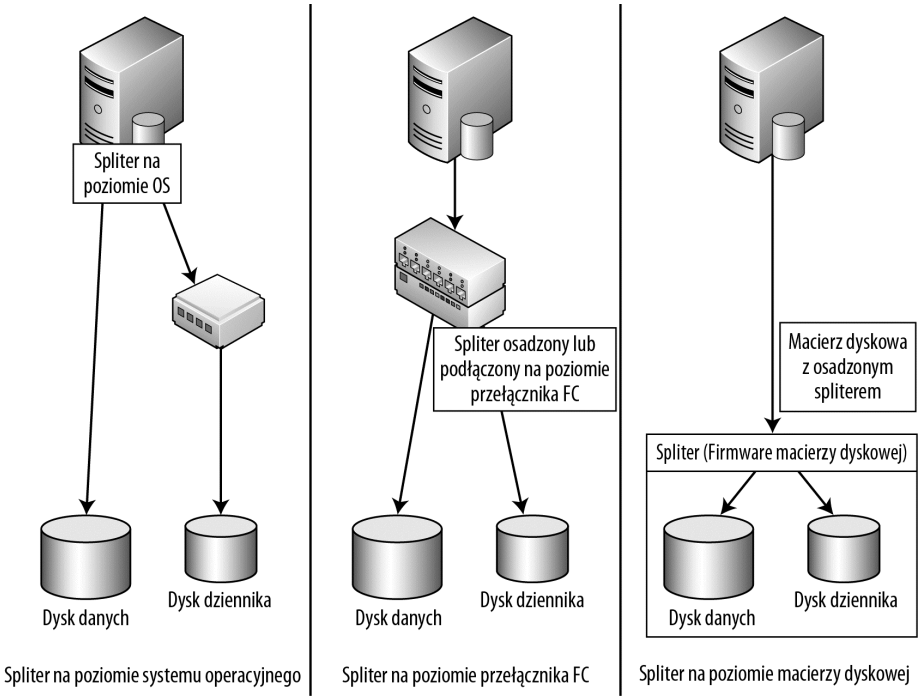
CDP w ścieżce operacji wejścia-wyjścia może być zaimplementowane na kilka sposobów. Najbardziej popularnymi punktami implementacji są serwery, infrastruktura pamięci SAN lub nawet wybrane macierze dyskowe. W przypadku implementacji na serwerze operacje wejścia-wyjścia są dzielone za pomocą sterownika rozdzielającego, który przechwytuje dane i przesyła je zarówno do dziennika, jak i podstawowej pamięci dyskowej serwera. Implementacja na poziomie SAN-a wymaga zazwyczaj dołączenia do przełącznika SAN specjalnego, zewnętrznego urządzenia spełniającego rolę splitera. Innym rozwiązaniem jest zainstalowanie splitera mającego postać specjalnego, dedykowanego modułu bezpośrednio w przełączniku. W implementacji na poziomie SAN-a żądania operacji wejścia-wyjścia są



Rysunek 5.15. Odzyskiwanie danych w metodzie CDP po napotkaniu błędu

wysyłane z serwera normalną drogą, zwykle za pośrednictwem kanału FC. Moduł CDP jest z logicznego punktu widzenia umieszczony pomiędzy pamięcią masową a serwerem, dzięki czemu może bez trudu przechwytywać żądania operacji wejścia-wyjścia wysyłane do pamięci masowej. Moduł CDP tworzy następnie odpowiedni wpis w dzienniku i przesyła oryginalne żądanie operacji do pamięci masowej. Ostatnim sposobem rozdzielania operacji wejścia-wyjścia jest umieszczenie splitera na poziomie macierzy dyskowej. LUN-y w takiej macierzy są przypisywane za pośrednictwem kontrolera macierzy i jeden z nich spełnia rolę dziennika (patrz rysunek 5.16).

Ciągła replikacja zdalna (CRR) to po prostu rozszerzenie CDP. Zamiast dziennika zapisywanego lokalnie CRR tworzy zarówno zdalny dziennik, jak i zdalną kopię danych. Dane są przesyłane i zapisywane w zdalnym dzienniku i następnie tworzona jest zdalna kopia danych, dzięki czemu zdalna replika jest gotowa do użycia. Jeżeli podczas tego procesu wystąpi błąd logiczny, aplikacja musi najpierw odwołać zdalną transakcję, a potem dokonać ponownej próby jej zapisu do dziennika i utworzenia zdalnej kopii danych. Po zatwierdzeniu transakcji dane są gotowe do użycia, choć fizycznie są zlokalizowane na zdalnym serwerze. Takie rozwiązanie jest zazwyczaj stosowane do przywracania funkcjonalności systemu po wystąpieniu awarii (ang. *DR — Distaster Recovery*) w celu zapewnienia maksymalnie



Rysunek 5.16. Porównanie metod implementacji CDP

krótkich czasów **RPO** (ang. *Recovery Point Objective*) oraz **RTO** (ang. *Recovery Time Objective*) odzyskiwania przynajmniej tych najbardziej krytycznych danych.

NetBackup udostępnia gotowe rozwiązanie CDP za pomocą pakietu RealTime — oprogramowania wykorzystującego sterownik rozdzielający na poziomie serwera. RealTime integruje się z pakietem NetBackup, pozwalając mu na zarządzanie tworzeniem obrazów danych w wybranych punktach czasu za pomocą mechanizmu CDP (taka kopia jest często nazywana migawką). Dzięki temu takie migawki mogą być traktowane jako główne kopie zapasowe i mogą być zapisywane na standardowych nośnikach pamięci masowych (na przykład na taśmach lub dyskach). Patrząc z drugiej strony, takie rozwiązanie pozwala również pakietowi NetBackup na traktowanie migawek wykonanych przez CDP jako podstawowych kopii zapasowych dla celów odzyskiwania danych na oryginalne serwery (w razie potrzeby dane z kopii zapasowych mogą być również odzyskiwane na inne serwery, na przykład w sytuacji, kiedy dana kopia zapasowa została już przeniesiona na nośniki pomocnicze).

CommVault nie posiada rozwiązania CDP z prawdziwego zdarzenia, aczkolwiek jego producent twierdzi, że pakiet posiada pewne możliwości systemów CDP. W praktyce jednak wspomniane możliwości sprowadzają się do uruchamiania w określonych odstępach czasu procesu replikacji danych z systemu plików na serwery pomocnicze.

Przechowywanie danych w chmurze

Kolejną, bardzo zaawansowaną technologią, która pojawiła się na rynku w ciągu ostatnich lat, jest chmura. Kiedy używamy określenia **chmura** (ang. *cloud*), bardzo istotny jest kontekst, w jakim go używamy. **Technologie przetwarzania w chmurze** mogą bowiem odnosić się do kilku zupełnie różnych elementów: pamięci masowej, aplikacji czy nawet chmur serwerów.

Najpierw zatem musimy omówić ogólną charakterystykę środowisk chmury. **Środowisko chmury** to środowisko lub zbiór zasobów, które z punktu widzenia użytkownika końcowego nie posiadają „widocznych" zasobów fizycznych, takich jak serwery czy pamięć masowa, są dostępne wyłącznie za pośrednictwem połączeń sieciowych (zwykle poprzez internet) i są dostępne w modelu „na życzenie" (ang. *on-demand*). Jest to środowisko w pewnym stopniu podobne do wirtualnych środowisk serwerowych, oferowanych przez takie produkty jak VMWare czy Microsoft Hyper-V, gdzie fizyczne podzespoły serwerów są oddzielone warstwą wirtualizacji od systemów operacyjnych gości. W przypadku chmury warstwa wirtualizacji oddziela cały system operacyjny lub inne rodzaje zasobów od użytkownika końcowego. Środowiska chmury mogą udostępniać użytkownikom dedykowane aplikacje, takie jak na przykład serwery Web, środowiska CRM czy bazy danych; mogą udostępniać całe środowiska serwerów wirtualnych lub po prostu oferować dostępne zdalnie zasoby pamięci masowych. Pamięć masowa dostępna w chmurze różni się od innych rozwiązań zdalnie udostępniających pamięć masową, takich jak NFS (ang. *Network File System*) czy CIFS (ang. *Common Internet File System*), kilkom istotnymi szczegółami:

- Pamięć masowa w chmurze nie udostępnia użytkownikowi widoku systemu plików czy struktury katalogów jak NFS czy CIFS. Zamiast tego wymaga programowego podejścia do zapisywania danych, które zamiast w plikach określonego typu są przechowywane po prostu w postaci obiektów binarnych.
- Protokół wykorzystywany do komunikacji z chmurą i zarządzania przechowywanymi w niej danymi jest bardzo uproszczony (nie wymaga angażowania dużej ilości zasobów). Najczęściej używanym protokołem komunikacji z chmurą jest REST, protokół oparty na protokole HTTP, który pozwala na przesyłanie danych bez nadmiernego obciążania sieci (co było domeną protokołów NFS i CIFS wymuszających sprawdzanie integralności danych).
- Pamięć masowa w chmurze posiada również relatywnie silny mechanizm uwierzytelniania użytkowników pragnących uzyskać dostęp do zasobów. Dzięki temu dane są chronione i dostęp mają do nich tylko uprawnieni użytkownicy.
- Chmura nie zapewnia jednak możliwości szyfrowania przesyłanych danych, stąd szyfrowanie danych, jeżeli jest wymagane, powinno się odbywać lokalnie, przed wysłaniem danych do chmury.
- W chmurze w zasadzie nie ma limitu rozmiarów pamięci masowej; zamiast tego użytkownik płaci za wykorzystaną przestrzeń dostawcy chmury.
- Pamięć masowa chmury jest optymalizowana do operacji typu „zapisz kilka razy, odczytuj do woli".

Zatem czy pamięć masowa w chmurze nadaje się do przechowywania kopii zapasowych? Z pewnością możemy z niej skorzystać przynajmniej dla kilku rodzajów operacji. Po pierwsze, możemy ją wykorzystywać jako pamięć masową do przechowywania danych przez długie okresy — możemy przesyłać do chmury kopie zapasowe przechowywane do tej pory na nośnikach tradycyjnych, zwłaszcza jeżeli są to nie tyle kopie zapasowe, co bardziej archiwa danych, czyli kopie zapasowe, których raczej nikt nie będzie szybko potrzebował, poza jakimiś specjalnymi okolicznościami. Przechowywanie w chmurze jest zwykle tańszym rozwiązaniem niż przechowywanie danych na nośnikach taśmowych, nie wspominając już o wielokrotnie większej niezawodności takiego rozwiązania.

Kolejnym zastosowaniem pamięci masowej w chmurze jest przechowywanie kopii zapasowych dedykowanych do celów odtwarzania funkcjonalności systemu po awarii i zapewnienia ciągłości działania firmy (ang. *Business Continuity and Disaster Recovery*). Ponieważ chmura bazuje przeważnie na internecie, jest z definicji dostępna z dowolnego

miejsca wyposażonego w odpowiednie łącze. Jeżeli wystąpi jakieś zdarzenie BC/DR, możemy odzyskać dane z chmury na dowolny serwer i w dowolnym miejscu na świecie — a nawet możemy odtworzyć dane na dowolny serwer w wirtualnej chmurze serwerów. Należy jednak pamiętać, że niektórzy dostawcy chmur obsługują tylko klientów z określonego obszaru czy kraju, stąd podczas projektowania rozwiązania należy wziąć pod uwagę również i takie elementy. Takie ograniczenia mogą mieć poważny wpływ na ostatnie zastosowanie pamięci masowej w chmurze: chmura może być wykorzystana do przenoszenia kopii zapasowych z jednej lokalizacji firmy do innej. Dzięki chmurze w przypadku zaistnienia zdarzenia BC/DR możemy szybko wysłać komplet danych do lokalizacji, która ucierpiała w zdarzeniu. Jako nowa technologia chmura podlega nieustannemu rozwojowi i niemal z dnia na dzień pojawiają się jej nowe możliwości i nowe potencjalne obszary zastosowań.

CommVault oferuje bardzo mocne wsparcie dla pamięci masowych w chmurze. Obiekt w chmurze spełnia rolę po prostu kolejnego serwera MagLib czy napędu taśmowego. NetBackup nie posiada co prawda bezpośredniego wsparcia dla chmury, ale może korzystać z danych tam przechowywanych za pośrednictwem bramek emulujących protokoły CIFS lub NFS.

Podsumowanie

Technologie przechowywania kopii zapasowych przeszły długą drogę, od taśm magnetycznych czy nawet od wirtualnych bibliotek taśmowych. Technologie takie jak deduplikacja, CDP czy pamięci masowe w chmurze zrewolucjonizowały rynek systemów do tworzenia kopii zapasowych i dostarczyły nowych metod ochrony dużych ilości danych oraz przechowywania ich w sposób, który nie byłby możliwy do osiągnięcia za pomocą tradycyjnych technologii.

Deduplikacja danych wprowadziła nową jakość do przechowywania danych na dyskach, powodując zredukowanie kosztów podstawowych rozwiązań dyskowych i sprowadzenie ich na poziom porównywalny z nośnikami taśmowymi. Wprowadzenie deduplikacji spowodowało również, że proces replikacji kopii zapasowych stał się dostępny zarówno z technologicznego, jak i ekonomicznego punktu widzenia oraz stał się realną alternatywą dla fizycznego transportu nośników taśmowych do odległych siedzib firmy. Deduplikacja może być zaimplementowana na wiele sposobów, zarówno po stronie źródła, jak i po stronie celu, co pozwala na wykorzystanie różnych metod analizy danych i zapisywania ich w pamięci masowej. Deduplikacja jest bardzo efektywnym sposobem zapisywania danych na dyskach, ale jednocześnie nie jest pozbawiona wad i nie sprawdza się w przypadku niektórych typów danych. Można zatem powiedzieć, że deduplikacja jest technologią o ogromnych możliwościach, która zrewolucjonizowała metody tworzenia kopii zapasowych oraz przechowywania danych.

Wprowadzenie technologii CDP oraz CRR do zastosowań związanych z tworzeniem kopii zapasowych pozwoliło na zredukowanie rozmiarów okien czasowych i skrócenie czasów odzyskiwania danych. Obie technologie udostępniły również możliwość efektywnej replikacji kopii zapasowych, co może pomóc w wyeliminowaniu potrzeby tworzenia dodatkowych obrazów kopii zapasowych na nośnikach taśmowych.

Wreszcie technologia chmury dokonała niespotykanej wcześniej wirtualizacji pamięci masowych. Kopie zapasowe mogą być przechowywane w zupełnie wirtualnym tworze, nie mającym żadnej reprezentacji fizycznej dla użytkownika końcowego — po prostu istnieją „gdzieś" w sieci (lokalizacja fizyczna przestaje mieć zatem jakiekolwiek znaczenie). Kiedy technologia chmury wyjdzie z początkowego okresu swojego rozwoju i stanie się dojrzałym, sprawdzonym rozwiązaniem, z pewnością zrewolucjonizuje świat kopii zapasowych, eliminując lokalne pamięci masowe na rzecz wirtualnych bytów bez określonej reprezentacji fizycznej.

Skorowidz

E

F

G

H

L

Ł

M

N

O

P

Q

R

S

Ś

T

U

V

W

Y

Z

BĄDŹ PEWNY, TWOJE DANE SĄ BEZPIECZNE!

Złość i bezsilność po utracie danych są nie do opisania. Każdy, kogo spotkało takie nieszczęście, z pewnością to potwierdzi. Strata danych może zachwiać niejedną firmą lub domem. Zgadza się — domem! Pomyśl tylko o utracie cyfrowych zdjęć z pierwszych urodzin Twojej pociechy. Celem tej książki nie jest straszenie, ale dostarczanie sprawdzonych strategii tworzenia kopii danych, istotnych zarówno z punktu widzenia firmy, jak i osoby prywatnej.

W trakcie lektury dowiesz się, na jakich nośnikach możesz wykonywać kopie danych, poznasz rodzaje macierzy dysków oraz ich zalety i wady. W kolejnych rozdziałach przeczytasz o zaawansowanych aplikacjach do tworzenia kopii bezpieczeństwa, takich jak Symantec NetBackup/BackupExec i CommVault Simpana, oraz różnych strategiach ich wykonywania. Nauczysz się tworzyć kopię baz danych (SQL Server, Oracle) oraz serwerów poczty (Exchange, Lotus Notes). Ponadto sprawdzisz, jak zweryfikować poprawność kopii, stworzyć raport z przeprowadzonego backupu oraz odtworzyć wybrane dane. Książka ta skupia się na zabezpieczaniu przed utratą danych w dużych środowiskach firmowych i korporacyjnych, jednak użytkownicy domowi, którym zależy na bezpie- czeństwie prywatnych informacji, także znajdą tu wiele cennych wskazówek. Jeśli los zawartości Twoich dysków nie jest Ci obojętny, przeczytaj tę książkę!

- Oprogramowanie do tworzenia kopii
- Nośniki danych — taśmy DLT, LTO i inne
- Dyski twarde i macierze RAID
- Pamięci dyskowe NAS i SAN
- Wirtualne nośniki danych
- Strategie tworzenia kopii zapasowych
- Archiwizacja baz danych oraz serwerów pocztowych
- Monitorowanie i raportowanie

Apress®

helion.pl księgarnia internetowa

Nr katalogowy: 7741

Księgarnia internetowa
http://helion.pl

Zamówienia telefoniczne:
0 801 339900
0 601 339900

Helion

Sprawdź najnowsze promocje:
http://helion.pl/promocje
Książki najchętniej czytane:
http://helion.pl/bestsellery
Zamów informacje o nowościach:
http://helion.pl/nowosci

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

Informatyka w najlepszym wydaniu

sięgnij po WIĘCEJ

KOD KORZYŚCI

ISBN 978-83-246-3417-0

Cena: 54,90 zł