Wzorce Operacyjne

Kamil Maksymowicz, Dawid Mazurkiewicz, Grzegorz Pryjma, Aleksandra Radziak, Krzysztof Rokosz, Maksymilian Siembab, Andrzej Sobierajski

KONCEPT

Poprawne zidentyfikowanie problemu

Definicja potrzebnych operacji do rozwiązania problemów

Wykonanie operacji w celu rozwiązania problemu

1
Plan rozwiązania problemu

2
Zaprogramowanie rozwiązania bez przejmowania się wykonanym rozwiązaniem

3
Optymalizacja problemu

Przykładowe wzorce operacyjne

Cache-aside

ładuje dane do pamięci cache z bazy danych na żądanie

Sharding

dzieli dane z baz danych w zestaw części

Index table

tworzy indeksy pomiędzy polami w bazie danych, które wywoływane są poprzez zapytania

Materialized view

generuje wstępnie wypełnione widoki danych w co najmniej jednej bazie danych, gdy dane nie są idealnie sformatowane dla operacji wymagających zapytań

Static content hosting

wdraża zawartość statyczną w oparciu o chmurę. Może dostarczać dane bezpośrednio do klienta.

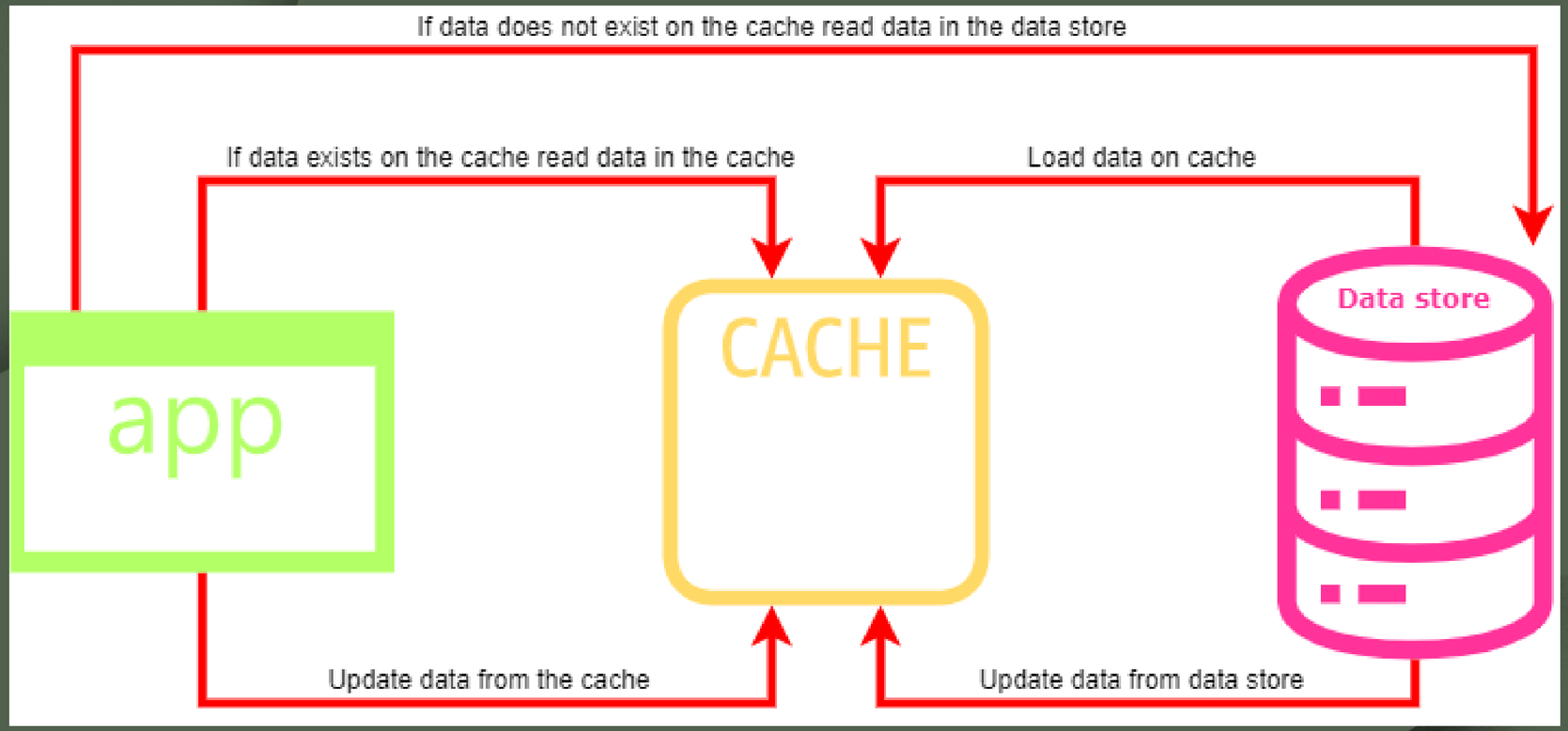
Event Sourcing

działa w trybie append-only. Rejestruje akcje podejmowane w domenie

CQRS

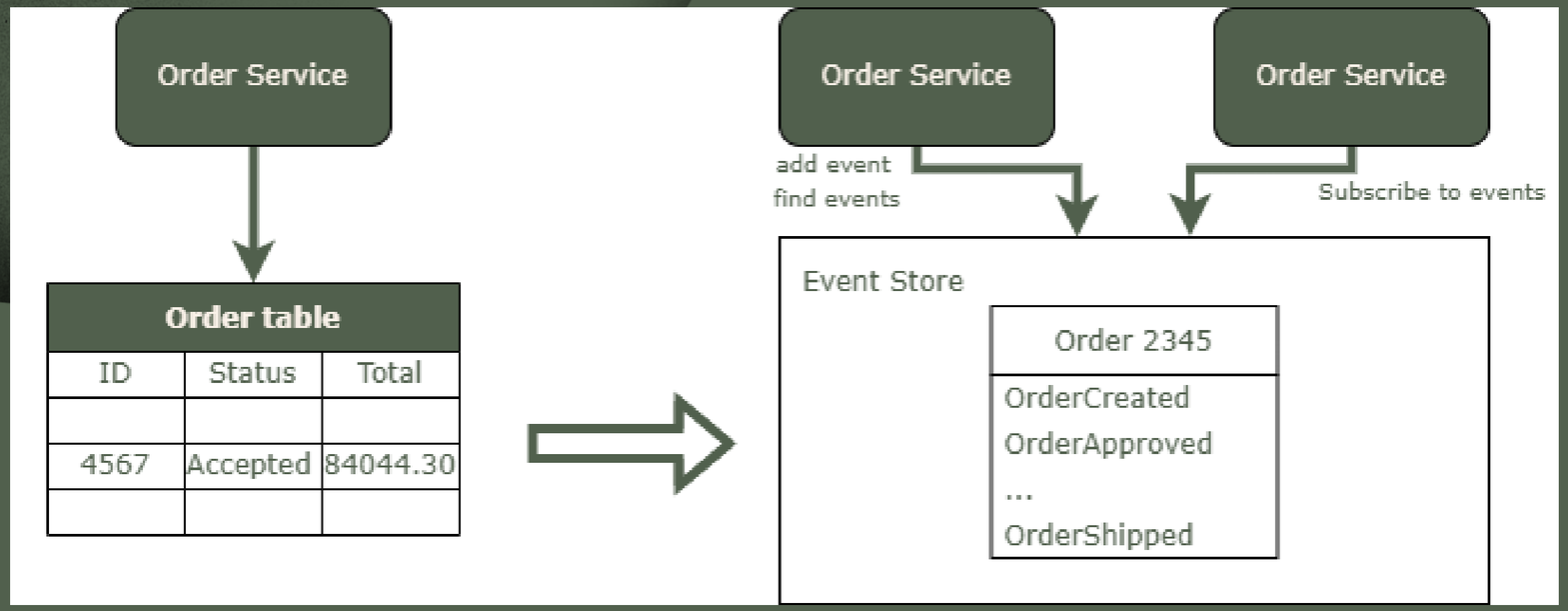
oddziela operacje, które odczytują dane z operacji, które aktualizują dane przy użyciu oddzielnych interfejsów.

Cache-aside



Wzorzec rekomendowany jest, gdy dane nie są często aktualizowane. Polega on na zapisaniu informacji do pamięci cache na żądanie i trzymanie informacji do czasu aż nie będzie potrzebna aktualizacja danych lub nie wygaśnie czas żywotności pamięci cache. Zaletą pamięci cache jest szybkość dostępu do pamięci i brak potrzeby ciągłego odpytywania bazy danych. W przypadku, gdy pamięć cache jest pusta dane są aktualizowane z bazy danych. Wadą wzorca jest to, że jeżeli dane zostaną zaktualizowanego z innego źródła możemy mieć różnice w pamięci cache i stanem faktycznym.

Event sourcing



To rozwiązanie gdzie zmiany danych są zapisywane do magazynu który przechowuje modyfikacje wykonane na danych jako ciąg operacji które należy wykonać na danych z magazynu aby odtworzyć z nich zmodyfikowane dane. Taki magazyn może służyć również jako historia zmian.

Event sourcing

Działanie magazynu Event Sourcing

1. Otrzymaj od jednego użytkownika serię poleceń modyfikujących dane
2. Dołącz otrzymane komendy do historii operacji w kolejności chronologicznej
3. Wyemituj wydarzenie o modyfikacji danych do wszystkich subskrybentów
4. Zwróć nowe polecenia wszystkim żądającym subskrybentom
5. Co jakiś czas wykonaj snapshot logu poleceń i wykonaj go na bazie danych, jednocześnie rozpoczynając nowy log zaczynający się od poleceń otrzymanych po snapshocie
6. Przenieś snapshot do archiwum

Index table

Wzorzec wykorzystywany do tworzenia indeksów dla tabel w bazie danych.

Wzorzec może poprawić wydajność zapytań, umożliwiając aplikacjom szybsze lokalizowanie danych do pobrania z bazy danych.

Daje możliwość odczytywania danych przy użyciu atrybutów, które nie zawierają indeksu jako filtru w zapytaniu.

Do konstruowania tabeli indeksów powszechnie stosuje się trzy strategie :

1. Strategia polegająca na zduplikowaniu danych w każdej tabeli indeksów, ale zorganizowaniu ich według różnych kluczy.

2. Tworzenie tabel indeksów uporządkowanych według różnych kluczy i odwoływanie się do oryginalnych danych przy użyciu klucza głównego zamiast jego duplikowania.

3. Strategia polegająca na tworzeniu częściowo znormalizowanych tabel indeksowych zorganizowanych według różnych kluczy, które duplikują często pobierane pola.

Index table

1

ID	Name	Age
1	Name A	20
2	Name B	22
3	Name C	20
4	Name D	19
5	Name E	22

Index(Age)	ID	Name	Age
19	4	Name D	19
20	1	Name A	20
20	3	Name C	20
22	2	Name B	22
22	5	Name E	22

2

ID	Name	Age
1	Name A	20
2	Name B	22
3	Name C	20
4	Name D	19
5	Name E	22

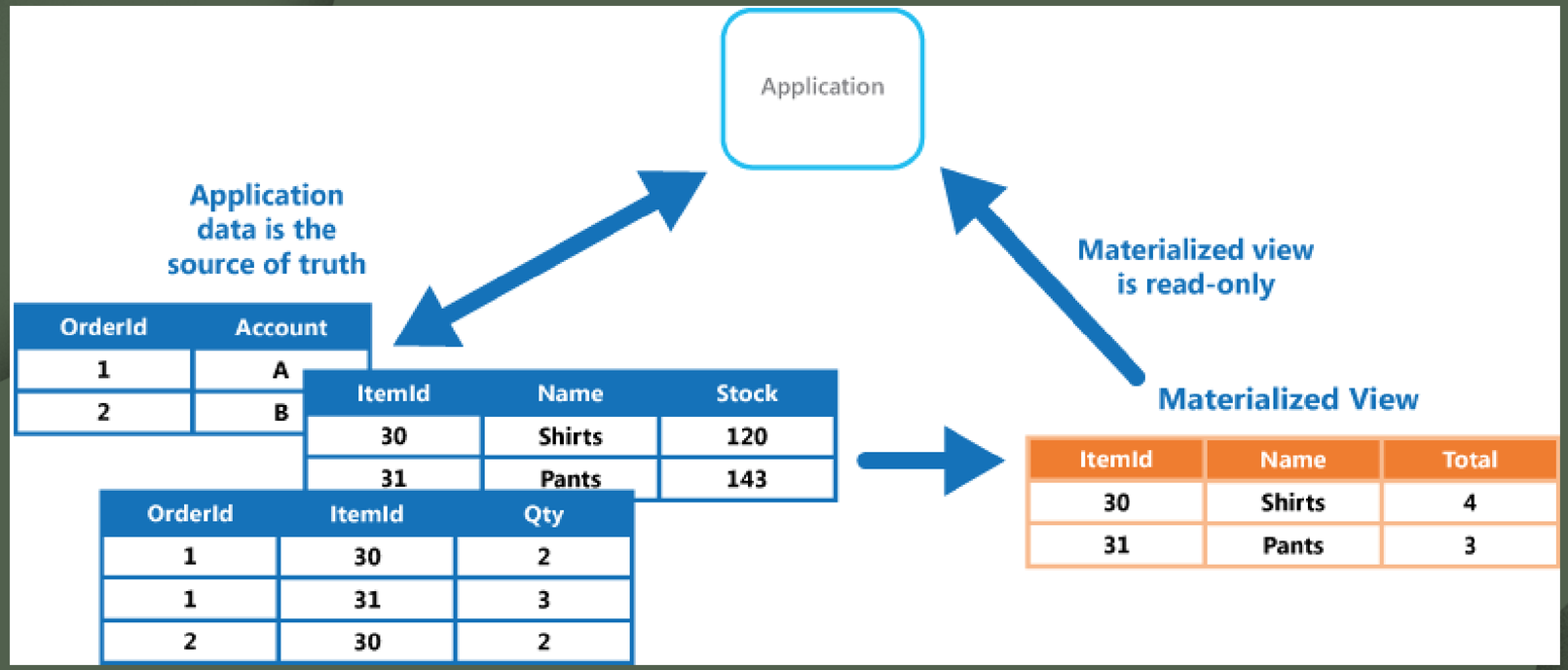
Index(Age)	ID
19	4
20	1
20	3
22	2
22	5

3

ID	Name	Age
1	Name A	20
2	Name B	22
3	Name C	20
4	Name D	19
5	Name E	22

Index(Age)	ID	Name
19	4	Name D
20	1	Name A
20	3	Name C
22	2	Name B
22	5	Name E

Materialized view



Czasami potrzebujemy zobaczyć dane połączone z różnych fizycznych lokalizacji. Często powoduje to problemy z wydajnością, spowalniając odczyt danych. W związku z tym, aby zwiększyć wydajność możemy utworzyć wstępnie wypełniony widok danych z wielu fizycznych lokalizacji - służy do tego wzorzec materialized view. Materialized view pattern charakteryzuje się tym, że wszelkie dane są odczytywane bez wykonywania łączeń lub obliczeń. Należy zauważyć, że dane uzyskane w taki sposób nigdy nie są aktualizowane w widoku. Gdy rzeczywiste dane zostaną zaktualizowane, widok będzie musiał zostać odbudowany. Dlatego zaleca się korzystanie z materialized view gdy dane są rzadko modyfikowane i nie są dynamiczne.

Sharding

Sharding jest to podzielenie magazynu danych na odłamki. Każdy taki odłamek zawierać będzie identyczną strukture co monolityczny magazyn ale wpisy są dzielone pomiędzy odłamkami. Każdy odłamek dostanie część danych zwykle opierający się na jakiejś charakterystyce danych np państwo z którego pochodzi zamówienie. Dane które posłużą nam do podzielenia monolita na odłamki staną się kluczem (shard key lub partiction key). Istnieją trzy zwykle stosowane metody wyboru takiego klucza

Wzorce Shardingu

Lookup strategy

W tej strategii implementowana jest mapa która kieruje żądanie do fragmentu zawierającego te dane przy użyciu klucza fragmentu. W aplikacji wielodostępnej wszystkie dane jednostki mogą być przechowywane razem we fragmencie przy użyciu identyfikatora jednostki jako klucza fragmentu. Wiele jednostek może współużytkować ten sam fragment, ale dane dla jednej jednostki nie zostaną rozłożone na wiele fragmentów.

Range strategy

W tej strategii implementowana jest metoda która umożliwia grupowanie powiązanych elementów, które są uporządkowane według klucza poszczególnego fragmentu. Klucze fragmentu w tej strategii są sekwencyjne. Jest to przydatne w przypadku aplikacji, które regularnie znajdują elementy za pomocą zapytań zwracających zestaw zakresu danych w poszczególnym fragmencie.

Hash strategy

W tej strategii implementowana jest metoda wykorzystująca funkcję skrótu, która umożliwia równomierne rozłożenie danych między fragmentami. Celem tej strategii jest zmniejszenie ryzyka nieproporcjonalnie obciążonych fragmentów.

Dziękujemy za uwagę!