

Spis treści

Wstęp	9
1. Myślenie grafowe	15
Dlaczego teraz? Kontekst technologii bazodanowych	16
Okres od lat 60. do lat 80. XX wieku — dane hierarchiczne	17
Okres od lat 80. XX wieku do pierwszej dekady XXI wieku — encja-relacja	18
Od początku XXI wieku do lat 20. XXI wieku — NoSQL	19
Lata 20. XXI wieku do ? — grafy	20
Czym jest myślenie grafowe?	22
Złożone problemy i złożone systemy	22
Problemy złożone w biznesie	23
Podejmowanie decyzji o technologii rozwiązywania złożonych problemów	25
Twoje dane są grafem. Co teraz?	28
Spojrzenie z szerszej perspektywy	31
Ruszamy na wyprawę z myśleniem grafowym	32
2. Ewolucja od myślenia relacyjnego do grafowego	33
Przegląd rozdziału: tłumaczenie koncepcji relacyjnych na terminologię grafów	33
Relacyjne kontra grafowe — na czym polega różnica?	34
Dane potrzebne w przykładzie	35
Modelowanie danych relacyjnych	36
Encje i atrybuty	37
Tworzenie diagramu ERD	37
Koncepcje związane z danymi grafowymi	38
Podstawowe elementy grafu	39
Przyleganie	40
Sąsiedztwa	40
Odległość	40
Stopień	41
Język Graph Schema Language	43
Etykiety wierzchołków i krawędzi	43
Właściwości	44
Kierunek krawędzi	45
Odwołujące się do siebie etykiety krawędzi	47

Mnogość grafu	48
Pełny przykładowy model grafu	50
Relacyjne kontra grafowe: decyzje do rozważenia	51
Modelowanie danych	51
Zrozumienie danych grafowych	52
Mieszanie projektu bazy danych z celem aplikacji	52
Podsumowanie	53
3. Zaczynamy. Prosta aplikacja Customer 360	55
Przegląd rozdziału: relacyjne kontra grafowe	56
Podstawowy przypadek użycia dla danych grafowych — C360	56
Dlaczego firmy przejmują się projektem C360?	57
Implementowanie aplikacji C360 w systemie relacyjnym	58
Modele danych	59
Implementacja relacyjna	61
Przykładowe zapytania dla aplikacji C360	65
Implementacja aplikacji C360 w systemie grafowym	68
Modele danych	68
Implementacja grafowa	69
Przykładowe zapytania C360	76
Relacyjne kontra grafowe — jak wybrać?	80
Relacyjne kontra grafowe — modelowanie danych	80
Relacyjne kontra grafowe — reprezentowanie relacji	80
Relacyjne kontra grafowe — języki zapytań	81
Relacyjne kontra grafowe — najważniejsze aspekty	82
Podsumowanie	82
Dlaczego nie relacyjne?	83
Wybór technologii dla aplikacji C360	83
4. Badanie sąsiedztwa w środowisku roboczym	85
Przegląd rozdziału — tworzenie bardziej realistycznej aplikacji Customer 360	85
Zasady modelowania danych grafowych	86
Czy to powinien być wierzchołek, czy krawędź?	87
Zgubiłeś się? Wskażemy Ci właściwy kierunek	89
Graf nie ma nazwy — typowe błędy w nazewnictwie	92
Gotowy model grafu w środowisku roboczym	94
Zanim zaczniemy budować	96
Nasze przemyślenia o znaczeniu danych, zapytań i użytkownika końcowego	96
Szczegóły implementacji eksploracji sąsiedztw w środowisku roboczym	97
Generowanie większej ilości danych dla rozszerzonego przykładu	98
Podstawowa nawigacja w języku Gremlin	99
Zaawansowane aspekty Gremlina — formatowanie wyników zapytania	106
Formatowanie wyników zapytania za pomocą kroków <code>project()</code> , <code>fold()</code> i <code>unfold()</code>	107
Usuwanie danych z wyników za pomocą wzorca <code>where(neq())</code>	110
Planowanie złożonych wyników za pomocą kroku <code>coalesce()</code>	111
Przejsięcie ze środowiska roboczego do produkcyjnego	114

5. Eksploracja sąsiedztw w środowisku produkcyjnym	115
Przegląd rozdziału — rozproszone dane grafowe w środowisku Apache Cassandra	116
Praca z danymi grafowymi w środowisku Apache Cassandra	117
Najważniejsze zagadnienie dotyczące modelowania danych — klucze główne	117
Klucze partycji i lokalizacja danych w środowisku rozproszonym	119
Opis krawędzi, część 1. Krawędzie na liście sąsiedztwa	123
Zrozumienie krawędzi, część 2. Kolumny klastrów	125
Zrozumienie krawędzi, część 3. Perspektywy zmaterializowane dla przejścia przez graf	129
Zaawansowane modelowanie danych grafowych	131
Znajdowanie indeksów za pomocą inteligentnego systemu rekomendacji indeksów	135
Szczegóły implementacji produkcyjnej	136
Perspektywy zmaterializowane i dodawanie czasu do krawędzi	136
Gotowy schemat produkcyjny aplikacji C360	138
Wczytywanie dużej ilości danych grafowych	139
Uzupełnianie zapytań w Gremlinie z wykorzystaniem czasu na krawędziach	142
Przejście do bardziej złożonych, rozproszonych problemów grafowych	144
10 pierwszych wskazówek dotyczących przejścia od środowiska roboczego do produkcyjnego	144
6. Używanie drzew w środowisku roboczym	147
Przegląd rozdziału — nawigowanie przez drzewa, dane hierarchiczne i cykle	147
Hierarchie i dane zagnieżdżone — trzy przykłady	148
Hierarchiczne dane w zestawieniu materiałów	148
Dane hierarchiczne w systemach kontroli wersji	148
Dane hierarchiczne w samoorganizujących się sieciach	149
Dlaczego stosuje się technologię grafową w przypadku danych hierarchicznych?	150
Jak się odnaleźć w lesie terminologii	150
Drzewa, korzenie i liście	151
Głębokość w przechodzeniu, ścieżki i cykle	152
Zrozumienie hierarchii w danych z czujników	154
Zrozumienie danych	154
Model koncepcyjny z wykorzystaniem notacji GSL	160
Implementowanie schematu	161
Zanim utworzymy zapytania	164
Zapytania wykorzystujące drogę od liści do korzeni w trybie roboczym	164
Dokąd wysłał dane określony czujnik?	165
Jaka jest droga od tego czujnika do dowolnej wieży?	168
Z dołu do góry	172
Przeszukiwanie od korzenia do liści w środowisku roboczym	172
Konfiguracja zapytania: jak znaleźć wieżę, z którą połączonych jest najwięcej czujników, aby można ją było wykorzystać w przykładzie?	173
Które czujniki są połączone bezpośrednio z wieżą Georgetown?	174
Szukanie wszystkich czujników połączonych z wieżą Georgetown	175
Ograniczanie głębokości w rekurencji	177
Powrót do przeszłości	178

7. Używanie drzew w środowisku produkcyjnym	179
Przegląd rozdziału — zrozumienie czynnika rozgałęziania i czasu na krawędziach	179
Zrozumienie czasu w danych dotyczących czujników	180
Ostatnie wnioski dotyczące danych serii czasowych w grafach	187
Zrozumienie czynnika rozgałęzień w naszym przykładzie	188
Czym jest czynnik rozgałęzień?	188
Jak sobie poradzić z czynnikiem rozgałęzień?	190
Schemat produkcyjny dla danych dotyczących czujników	190
Zapytania przechodzące od liści do korzeni w środowisku produkcyjnym	192
Dokąd i kiedy czujnik wysłał dane?	192
Znajdź wszystkie drzewa prowadzące od czujnika do wieży z uwzględnieniem czasu	193
Znajdź poprawne drzewo wychodzące z określonego czujnika	195
Zaawansowane aspekty Gremlina — wzorec <code>where().by()</code>	197
Zapytania od korzeni do liści w środowisku produkcyjnym	199
Które czujniki są bezpośrednio połączone z wieżą Georgetown według czasu?	199
Jakie prawidłowe drogi wiodą od wieży Georgetown w dół do wszystkich czujników?	200
Wykorzystanie zapytań w scenariuszach awarii wieży	204
Wykorzystanie ostatnich wyników do rozwiązania złożonego problemu	208
Dostrzeganie lasu w grupie drzew	208
8. Szukanie dróg w środowisku roboczym	209
Podgląd rozdziału — ocena ilościowa zaufania w sieciach	209
Myślenie o zaufaniu — trzy przykłady	210
Jak bardzo ufasz temu zaproszeniu?	210
Jak obronić historię śledczego?	211
Jak firmy modelują dostarczanie paczek?	212
Fundamentalne koncepcje dotyczące dróg	213
Najkrótsze drogi	213
Przeszukiwanie w głąb i przeszukiwanie wszcz	215
Nauka postrzegania cech aplikacji jako różnych problemów przeszukiwania dróg	216
Szukanie dróg w sieci zaufania	217
Dane źródłowe	217
Krótkie wprowadzenie do terminologii związanej z Bitcoinem	218
Tworzenie schematu roboczego	219
Wczytywanie danych	220
Analiza społeczności zaufania	220
Zrozumienie przejść w sieci zaufania Bitcoina	222
Które adresy znajdują się w pierwszym sąsiedztwie?	222
Które adresy znajdują się w drugim sąsiedztwie?	223
Które adresy znajdują się tylko w drugim sąsiedztwie?	224
Strategie wartościowania w języku zapytań Gremlin	225
Wybór losowego adresu do użycia w przykładzie	226
Zapytania wyszukujące najkrótsze drogi	227
Znajdowanie dróg o ustalonej długości	227
Znajdowanie dróg o dowolnej długości	229
Uzupełnianie dróg wartościami zaufania	232
Czy ufasz tej osobie?	238

9. Znajdowanie dróg w środowisku produkcyjnym	239
Przegląd rozdziału — zrozumienie wag, odległości i przycinania	239
Ważone drogi i algorytmy wyszukiwania	240
Definicja problemu najkrótszych dróg ważonych	240
Techniki optymalizacji przeszukiwania najkrótszych dróg ważonych	241
Normalizacja wag krawędzi dla problemów dotyczących najkrótszej drogi	244
Normalizacja wag krawędzi	245
Aktualizacja grafu	249
Eksploracja znormalizowanych wag krawędzi	250
Przemyślenia przed utworzeniem zapytań wyszukujących najkrótszą drogę ważoną	253
Zapytania o najkrótszą drogę ważoną	254
Tworzenie produkcyjnej wersji zapytania o najkrótszą drogę ważoną	254
Drogi ważne i zaufanie w środowisku produkcyjnym	263
10. Rekomendacje w środowisku roboczym	265
Przegląd rozdziału — kolaboratywne filtrowanie rekomendacji filmów	265
Przykłady systemów rekomendacji	266
Rekomendacje w służbie zdrowia	266
Na czym polegają rekomendacje w serwisach społecznościowych	267
Wykorzystanie głęboko połączonych danych do tworzenia rekomendacji w handlu elektronicznym	268
Wstęp do filtrowania kolaboratywnego	269
Zrozumienie problemu i domeny	269
Filtrowanie kolaboratywne danych grafowych	270
Rekomendacje na podstawie filtrowania kolaboratywnego opartego na elemencie zastosowanego w danych grafowych	271
Trzy różne modele tworzenia rankingu rekomendacji	272
Dane dotyczące filmów — schemat, wczytywanie i zapytania	276
Model danych dla rekomendacji filmów	276
Kod schematu dla rekomendacji filmów	277
Wczytywanie danych filmów	279
Zapytania dotyczące sąsiedztw w danych o filmach	283
Zapytania wykorzystujące drzewa w celu analizy danych o filmach	285
Zapytania przeszukujące drogi w danych o filmach	287
Filtrowanie kolaboratywne oparte na elementach w Gremlinie	289
Model 1. Liczenie dróg w zbiorze rekomendacji	289
Model 2. Zainspirowany NPS	290
Model 3. Znormalizowana punktacja NPS	292
Wybór swojej przygody — filmy i edycja problemu grafowego	294
11. Proste łączenie encji w grafach	295
Przegląd rozdziału — scalanie wielu zbiorów danych w jeden graf	295
Definiowanie innego złożonego problemu — łączenie encji	296
Analiza złożonego problemu	297
Analiza dwóch zbiorów danych o filmach	298
Zbiór danych MovieLens	299
Zbiór danych Kaggle	304
Schemat roboczy	307

Dopasowywanie i scalanie danych o filmach	308
Proces dopasowywania	308
Rozwiązywanie wyników fałszywie pozytywnych	310
Elementy fałszywie pozytywne w zbiorze danych MovieLens	311
Dodatkowe błędy wykryte podczas łączenia encji	311
Ostatnia analiza procesu scalania	313
Rola struktury grafu w scalaniu danych o filmach	313
12. Rekomendacje w środowisku produkcyjnym	315
Przegląd rozdziału — zrozumienie krawędzi skrótowych, wstępne obliczenia i zaawansowane obcinanie	316
Krawędzie skrótowe do ustalania rekomendacji w czasie rzeczywistym	316
Gdzie proces roboczy się nie skaluje	317
Obsługa problemów ze skalowaniem — krawędzie skrótowe	318
Analiza funkcjonalności w środowisku produkcyjnym	318
Przycinanie — różne sposoby wstępnego obliczania krawędzi skrótowych	319
Czynniki, jakie trzeba uwzględnić podczas aktualizacji rekomendacji	321
Obliczanie krawędzi skrótowych dla danych o filmach	322
Podział złożonego problemu wstępnego obliczania krawędzi skrótowych	322
Radzenie sobie ze słoniem w składzie porcelany — obliczenia masowe	326
Schemat produkcyjny i wczytywanie danych dla rekomendacji filmów	328
Schemat produkcyjny dla rekomendacji filmów	328
Wczytywanie danych produkcyjnych dla rekomendacji filmów	329
Zapytania dotyczące rekomendacji wykorzystujące krawędzie skrótowe	330
Potwierdzenie poprawnego wczytania krawędzi	331
Rekomendacje dla użytkownika w środowisku produkcyjnym	332
Zrozumienie czasu odpowiedzi w środowisku produkcyjnym poprzez zliczanie partycji krawędzi	336
Ostatnie uwagi dotyczące analizy wydajności rozproszonych zapytań grafowych	338
13. Epilog	339
Co dalej?	340
Algorytmy grafowe	340
Grafy rozproszone	341
Teoria grafów	341
Teoria sieci	342