

Wstęp (9)

1. Widok z góry (19)

- Co to znaczy "enterprise?" (19)
- Powolny wzrost (21)
- Komponenty aplikacji (24)
 - o Warstwa danych (24)
 - o Warstwa aplikacyjna (26)
 - o Warstwa cache'owania (29)
 - o System komunikacyjny (32)
 - o Serwer WWW (33)
 - o Zapora sieciowa (33)

2. Wtyczki jako środek organizacji kodu (35)

- Korzyści (36)
- Tworzenie własnej wtyczki (37)
 - o Rozszerzanie klas wbudowanych (38)
 - o Rozszerzenia uniwersalne (40)
- Wdrażanie (45)
 - o svn:externals (45)

3. Organizacja kodu za pomocą modułów (47)

- Pliki i katalogi (47)
- Granice modułu wyznaczają przestrzeń nazw (49)
 - o Międzymodułowe skojarzenia klas modelowych (50)
 - o Relacje wzajemne (51)
 - o Modularyzacja jako wstęp do architektury usługowej (51)
 - o Wymuszenie prawidłowej kolejności ładowania definicji klas (53)
- Ćwiczenia (54)
- Refaktoring (54)
 - o Wyodrębnianie modułów fizycznych (54)
 - o Uwalnianie metod użytkowych (55)

4. Baza danych jak forteca (57)

- Baza danych jako część aplikacji (58)
 - o "Jedno środowisko wyznacza reguły" (58)
 - o "Nasi programiści nie popełniają błędów" (58)
 - o "Baza danych moja i tylko moja" (59)
- Siadając na ramionach gigantów (59)
 - o Wybór właściwego systemu bazy danych (59)
 - o A propos migracji (60)
 - o Obalając mity... (62)
- Raporty, raporty... (63)

5. Budowanie solidnego modelu danych (67)

- Bilety do kina (67)

- o Na początek bardzo prosto (68)
- o Ograniczenia (70)
- o Obalamy mity (78)
- o Integralność referencyjna (78)
- o Wprowadzenie do indeksowania (85)

6. Refaktoryzacja bazy do trzeciej postaci normalnej (87)

- Trzecia postać normalna (87)
 - o Zaczniij od normalizacji (91)
- Dziedziczenie tabel i domieszki (92)
- Ćwiczenia (95)
- Refaktoryzacja (96)

7. Dane dziedziczne (97)

- Kody pocztowe i geograficzne dane dziedziczne (99)
- Wzorzec projektowy - strategia dla tabel dziedzicznych (101)
- Refaktoryzacja od samego początku (104)

8. Klucze złożone i postać normalna DKNF (107)

- Klucze naturalne - korzyści i kłopoty (108)
 - o Wybór kluczy naturalnych (111)
- Siedząc już na ramionach giganta... (112)
 - o Migracja do postaci normalnej DKNF (113)
 - o Klucze wielokolumnowe i ich implementacja w Rails (116)
 - o Odroczonego kontrola integralności referencyjnej (120)
 - o Coś za coś... (122)
- Ćwiczenia (123)
- Refaktoryzacja (124)
 - o Klucz jednokolumnowy (124)
 - o Klucz wielokolumnowy (125)

9. Wyzwalacze jako narzędzia kontroli skomplikowanych zależności wewnątrz danych (127)

- Kontrola ograniczeń za pomocą wyzwalaczy (127)
- Anatomia funkcji w języku PL/pgSQL (130)
 - o To tylko łańcuchy... (131)
 - o Zmienne lokalne i przypisywanie im wartości (131)
 - o Bloki (132)
 - o Dodatkowe cechy wyzwalacza (132)
 - o Wyzwalacz - łagodna zaporę lub bezpiecznik (132)
 - o Instrukcje warunkowe (133)

10. Dziedziczenie wielotabelaryczne (135)

- O co chodzi? (135)
- Polimorfizm - co to jest? (137)

- Dziedziczenie a dane fizyczne (138)
 - o Dziedziczenie jednotabelaryczne (140)
 - o Dziedziczenie wielotabelaryczne (140)
 - o Alternatywa wyłączająca dla zbioru kolumn (143)
 - o Implementacja MTI w Rails (145)
- Klasy-fabryki (151)
- Ćwiczenia (152)
- Refaktoryzacja (152)
 - o Z STI do MTI (152)
 - o Z :polymorphic => true do MTI (153)

11. Modele widokowe (155)

- Widoki (156)
 - o Definiowanie widoku (156)
- Definiowanie klasy modelowej na bazie widoku (157)
- Specyfika widoków (158)
 - o Dodawanie, modyfikowanie i usuwanie rekordów (159)
 - o Ograniczenia i klucze obce (159)
 - o Indeksowanie (160)
- Ćwiczenia (161)
- Refaktoryzacja (161)

12. Widoki zmaterializowane (163)

- Reguły rządzące widokami zmaterializowanymi (164)
- Widok źródłowy (165)
 - o Formatowanie widoku (166)
- Tabela docelowa (168)
- Funkcje odświeżające i unieważniające (169)
 - o Zarządzanie zależnościami czasowymi (171)
 - o Kto za to płaci? (172)
- Odświeżanie i unieważnianie sterowane wyzwalaczami (175)
 - o Tabela movie_showtimes (176)
 - o Tabela movies (178)
 - o Tabela theatres (178)
 - o Tabela orders (179)
 - o Tabela purchased_tickets (180)
- Ukrycie implementacji dzięki widokowi uzgadniającemu (181)
 - o Periodeczne odświeżanie (183)
- Indeksowanie widoku zmaterializowanego (184)
- To się naprawdę opłaca... (185)
- Kaskadowe cache'owanie widoków (186)
- Ćwiczenia (187)

13. SOA - zaczynamy (189)

- Czym jest SOA? (189)
- Dlaczego SOA? (192)
 - o Współdzielenie zasobów (193)

- o Redukcja obciążenia baz danych (196)
 - o Skalowalność i cache'owanie (202)
 - o Lokalna redukcja złożoności (202)
- Podsumowanie (205)
- Ćwiczenia (205)

14. Specyfika SOA (207)

- Specyfika usług (207)
 - o Ukryta implementacja (207)
 - o Przystępne API (210)
- Projektowanie API (211)
 - o Nie rozdrabniaj się (211)
 - o Ogranicz kontakty (213)
 - o Korzystaj ze współbieżności (214)
 - o Tylko to - i nic więcej (215)
- REST, XML-RPC i SOAP (217)
 - o XML-RPC (217)
 - o SOAP (219)

15. Usługi typu XML-RPC (221)

- ActionWebService w Rails 2.0 (221)
- Definiowanie bariery abstrakcji (222)
 - o ActiveRecord jako warstwa modelu fizycznego (222)
 - o Warstwa modelu logicznego (224)
 - o Definiowanie API (229)
- Więcej testów... (233)
- Wtyczka kliencka (235)
 - o Współdzielony kod (236)
 - o Kliencka klasa-singleton (237)
 - o Testy integracyjne (238)

16. Przechodzimy na SOA (241)

- Usługa zamówień - OrdersService (242)
- Integracja z usługą MoviesService (252)
 - o Konsekwencje... (254)
- Model obiektowy usługi MoviesService (256)
- Podsumowanie (265)

17. Usługi typu REST (267)

- Podstawy REST (267)
 - o Zasoby i polecenia (267)
 - o Sprzęt jest częścią aplikacji (269)
- REST a SOA (270)
 - o REST a CRUD (270)
 - o Uniwersalny interfejs (271)
 - o HTTP+POX (273)

- o Definiowanie kontraktu usługi (274)
- o Klient REST w Rails (276)
- o REST czy XML-RPC? (276)

18. Usługi webowe typu RESTful (279)

- Sformułowanie zadania (279)
- Narzędzia (281)
 - o ROXML (281)
 - o Net::HTTP (283)
- Usługa MoviesWebService (284)
 - o Implementacja zasobów serwera (284)
 - o Implementacja akcji serwera (287)
 - o Implementacja klienta (288)

19. Cache'owanie (295)

- Dla przypomnienia - cache'owanie w warstwie fizycznej (296)
 - o Migawki (296)
 - o Funkcja odświeżająca (297)
 - o Wyzwalacze unieważniające (297)
 - o Indeksowanie (298)
- Cache'owanie modeli logicznych (298)
 - o Uwarunkowania (304)
 - o Pułapka nieaktualnych danych (307)
 - o Indeksowanie cache (310)
- Inne aspekty cache'owania (311)
 - o Cache'owanie planu realizacji (311)
 - o Cache'owanie żądań (312)
 - o Cache'owanie w Rails (313)
 - o Cache'owanie fragmentów, akcji i stron (314)

Skorowidz (315)