

Słowo wstępne (9)

Przedmowa (11)

Podziękowania (13)

O książce (17)

1. Czym jest chmura obliczeniowa? (25)

- 1.1. Pięć podstawowych zasad definiujących przetwarzanie w chmurze (27)
 - o 1.1.1. Pula zasobów (28)
 - o 1.1.2. Wirtualizacja zasobów obliczeniowych (29)
 - o 1.1.3. Elastyczność wobec zmieniającego się zapotrzebowania (30)
 - o 1.1.4. Automatyczne wdrażanie nowych zasobów (30)
 - o 1.1.5. Naliczanie opłat: płacisz tylko za to, co faktycznie wykorzystasz (31)
- 1.2. Zyski z przejścia na chmurę (31)
 - o 1.2.1. Zyski ekonomiczne związane z zamianą wydatków inwestycyjnych na operacyjne (31)
 - o 1.2.2. Zyski związane z elastycznością i brakiem zapotrzebowania na serwery (32)
 - o 1.2.3. Zyski wydajnościowe dające przewagę nad konkurencją (33)
 - o 1.2.4. Większe bezpieczeństwo w chmurze (33)
- 1.3. Ewolucja w informatyce prowadząca do chmury obliczeniowej (33)
 - o 1.3.1. Dlaczego "chmura"? (34)
 - o 1.3.2. Zmiany paradygmatów przetwarzania: od samodzielnych jednostek, przez architektury klient-serwer, aż do sieci (35)
 - o 1.3.3. Przechowywanie fizycznych zasobów obliczeniowych: ewolucja centrów danych (37)
 - o 1.3.4. Modularyzacja oprogramowania i zdalny dostęp: wirtualizacja, SOA i SaaS (37)
- 1.4. Klasyfikacja warstw chmury: różne typy do różnych zastosowań (38)
 - o 1.4.1. Infrastruktura jako usługa (IaaS) (39)
 - o 1.4.2. Platforma jako usługa (PaaS) (41)
 - o 1.4.3. Oprogramowanie jako usługa (SaaS) i framework jako usługa (FaaS) (41)
 - o 1.4.4. Chmury prywatne jako prekursorzy chmur publicznych (42)
- 1.5. Podsumowanie (42)

2. Klasyfikacja chmur obliczeniowych (43)

- 2.1. Podstawy technologiczne przetwarzania w chmurze (44)
 - o 2.1.1. Duże korzyści skali dzięki centrom danych w chmurze (45)
 - o 2.1.2. Efektywne wykorzystanie serwerów w chmurze dzięki wirtualizacji (49)
 - o 2.1.3. Sterowanie zdalnymi serwerami za pośrednictwem API chmury (52)
 - o 2.1.4. Przechowywanie trwałych danych w chmurze (54)
 - o 2.1.5. Przechowywanie danych aplikacji w chmurowej bazie danych (56)
 - o 2.1.6. Elastyczność: skalowanie aplikacji w miarę zwiększania się lub zmniejszania popytu (62)
- 2.2. Zrozumienie różnych typów chmur (63)
 - o 2.2.1. Amazon EC2: IaaS (64)
 - o 2.2.2. Microsoft Azure: IaaS (65)
 - o 2.2.3. Google App Engine: PaaS (68)
 - o 2.2.4. Ruby on Rails w chmurze: PaaS (69)

- o 2.2.5. Salesforce.com i Force.com: PaaS (70)
 - o 2.2.6. Chmury prywatne: DaaS (centrum danych jako usługa) (70)
- 2.3. Wybór chmury najlepiej dopasowanej do Twoich potrzeb (72)
 - o 2.3.1. Amazon Web Services - chmura IaaS (72)
 - o 2.3.2. Microsoft Azure - chmura IaaS i PaaS (73)
 - o 2.3.3. Google App Engine - chmura PaaS (74)
 - o 2.3.4. Ruby on Rails - chmura PaaS (74)
 - o 2.3.5. Force.com - chmura PaaS (75)
- 2.4. Podsumowanie (75)

3. Analiza biznesowa chmury (77)

- 3.1. Ekonomia przetwarzania w chmurze (78)
 - o 3.1.1. Tradycyjna infrastruktura wewnętrzna, kolokacja, usługi zarządzane, a może model chmury? (79)
 - o 3.1.2. Szczegółowe porównanie kosztów wdrażania w różnych modelach (81)
- 3.2. Kiedy wdrożenie w chmurze ma sens? (86)
 - o 3.2.1. Ograniczony czas życia lub zapotrzebowanie krótkoterminowe (87)
 - o 3.2.2. Wahnięcia skali (88)
 - o 3.2.3. Aplikacje niestrategiczne (89)
- 3.3. Kiedy wdrożenie w chmurze nie ma sensu? (90)
 - o 3.3.1. Historyczne aplikacje (90)
 - o 3.3.2. Aplikacje z krytycznymi scenariuszami czasu rzeczywistego (91)
 - o 3.3.3. Aplikacje z dostępem do poufnych danych (91)
- 3.4. Przedsiębiorstwa typu start-up bez kapitału zakładowego (92)
 - o 3.4.1. Wtedy i teraz: tworzenie niewielkiego sklepu internetowego w 2000 i 2010 roku (92)
 - o 3.4.2. Czy zewnętrzny kapitał inwestycyjny jest niezbędny? (93)
 - o 3.4.3. Przykład 1.: FlightCaster - przewidywanie opóźnień lotów (94)
 - o 3.4.4. Przykład 2.: analiza biznesowa jako SaaS (94)
- 3.5. Małe i średnie przedsiębiorstwa (95)
 - o 3.5.1. Prosty przykład: strona firmowa (95)
 - o 3.5.2. Średnio skomplikowany przykład: kopie zapasowe i przechowywanie plików (96)
 - o 3.5.3. Przykład zaawansowany: rozwijanie nowych produktów (96)
- 3.6. Chmura w korporacjach (97)
 - o 3.6.1. Eli Lilly: duży zbiór danych, obliczenia wysokowydajne (97)
 - o 3.6.2. "The Washington Post": duże problemy obliczeniowe z nieprzekraczalnymi terminami (98)
 - o 3.6.3. Virgin Atlantic: obecność w sieci i zgromadzenie społeczności (99)
- 3.7. Podsumowanie (99)

4. Bezpieczeństwo i chmura prywatna (101)

- 4.1. Bezpieczeństwo informacji w chmurze publicznej (102)
 - o 4.1.1. Obawy o bezpieczeństwo spowalniające ekspansję chmury (103)
 - o 4.1.2. Bezpieczeństwo największych centrów danych w chmurze (104)
 - o 4.1.3. Środki kontroli dostępu w chmurze publicznej (106)
 - o 4.1.4. Bezpieczeństwo sieciowe i bezpieczeństwo danych w dużych chmurach (111)

- o 4.1.5. Rola i zakres odpowiedzialności właściciela aplikacji (114)
- 4.2. Przyczyny powstania chmury prywatnej (115)
 - o 4.2.1. Definicja chmury prywatnej (115)
 - o 4.2.2. Kwestie bezpieczeństwa (117)
 - o 4.2.3. Pewność dostępności zasobów (117)
 - o 4.2.4. Duża społeczność (118)
 - o 4.2.5. Efekty skali (118)
 - o 4.2.6. Potencjalne problemy z chmurą prywatną (119)
 - o 4.2.7. Sposoby wdrożenia chmury prywatnej (119)
- 4.3. Wirtualna chmura prywatna (124)
 - o 4.3.1. Jak to działa? (124)
 - o 4.3.2. API wirtualnej chmury prywatnej (125)
 - o 4.3.3. Konsekwencje (126)
- 4.4. Chmury prywatne w praktyce (126)
 - o 4.4.1. Sprint: chmura prywatna dla aplikacji wykrywającej oszustwa (127)
 - o 4.4.2. Project Services Network (PSN) firmy Bechtel (127)
 - o 4.4.3. Rządowe chmury prywatne (128)
- 4.5. Długoterminowa prognoza dla chmury prywatnej (129)
- 4.6. Podsumowanie (130)

5. Projektowanie i architektura aplikacji w chmurze (131)

- 5.1. Wzorce aplikacji najlepiej pasujące do chmury (132)
 - o 5.1.1. Przeniesienie (132)
 - o 5.1.2. Skala internetowa (133)
 - o 5.1.3. Ekspansja obliczeń (133)
 - o 5.1.4. Elastyczne składowanie danych (134)
 - o 5.1.5. Podsumowanie wzorców aplikacji (134)
- 5.2. Projektowanie i architektura w skali internetowej: shardowanie (134)
 - o 5.2.1. Cechy aplikacji blokujące skalowalność (136)
 - o 5.2.2. Shardowanie: zrównoleglona architektura bazy danych umożliwiająca skalowanie (137)
 - o 5.2.3. Jak shardowanie zmienia aplikację (139)
 - o 5.2.4. Porównanie shardowania z tradycyjnymi architekturami baz danych (140)
 - o 5.2.5. Shardowanie w praktyce: najpopularniejsze schematy partycjonowania baz danych (143)
 - o 5.2.6. Trudności i problemy związane ze shardowaniem (145)
 - o 5.2.7. Shardowanie w praktyce: jak robi to Flickr? (148)
- 5.3. Zwiększenie mocy na życzenie: cloudbursting (150)
 - o 5.3.1. Cloudbursting: definicja (150)
 - o 5.3.2. Dwie pieczenie na jednym ogniu: wewnętrzne centrum danych oraz chmura (151)
 - o 5.3.3. Cloudbursting: analiza biznesowa (152)
 - o 5.3.4. Cloudbursting: architektura (154)
 - o 5.3.5. Jak zaimplementować cloudbursting? (156)
 - o 5.3.6. Cloudbursting: potrzeba standaryzacji (157)
 - o 5.3.7. Cloudbursting: problem dostępu do danych (157)
- 5.4. Jak przygotować się na wykładniczy przyrost ilości składowanych danych? (160)
 - o 5.4.1. Magazyn danych w chmurze: definicja (160)

- o 5.4.2. Amazon S3 (161)
- o 5.4.3. Przykładowy interfejs magazynu danych w chmurze (S3) (161)
- o 5.4.4. Koszty (164)
- o 5.4.5. Montowalne systemy plików w chmurze (164)
- o 5.4.6. Jak sobie radzić z opóźnieniami? (165)
- 5.5. Podsumowanie (166)

6. Niezawodność w skali chmury (167)

- 6.1. SOA jako prekursor chmury (168)
 - o 6.1.1. Systemy rozproszone (168)
 - o 6.1.2. Luźne sprzężenie (170)
 - o 6.1.3. SOA (172)
 - o 6.1.4. SOA i luźne sprzężenie (173)
 - o 6.1.5. SOA i usługi sieciowe (174)
 - o 6.1.6. SOA i przetwarzanie w chmurze (175)
 - o 6.1.7. Komunikacja między procesami w chmurze (176)
- 6.2. Niezawodność wysokowydajnych, rozproszonych aplikacji w chmurze (176)
 - o 6.2.1. Nadmiarowość (177)
 - o 6.2.2. MapReduce (178)
 - o 6.2.3. Hadoop: MapReduce w wersji open source (183)
- 6.3. Podsumowanie (184)

7. Testy, wdrożenie i działanie w chmurze (185)

- 7.1. Typowe wdrożenia (186)
 - o 7.1.1. Tradycyjna architektura wdrożeniowa (187)
 - o 7.1.2. Środowisko testowe i środowisko etapu pośredniego (188)
 - o 7.1.3. Wyliczenie kosztów (189)
- 7.2. Chmura na ratunek! (189)
 - o 7.2.1. Poprawa jakości produkcyjnej dzięki chmurze (190)
 - o 7.2.2. Szybsze wytwarzanie aplikacji oraz testowanie (192)
- 7.3. Siła równoległości (195)
 - o 7.3.1. Testy jednostkowe (196)
 - o 7.3.2. Testy funkcjonalne (198)
 - o 7.3.3. Testy obciążeniowe (201)
 - o 7.3.4. Testy wizualne (204)
 - o 7.3.5. Testy ręczne (206)
- 7.4. Podsumowanie (207)

8. Kwestie praktyczne (209)

- 8.1. Wybór dostawcy chmury (210)
 - o 8.1.1. Kwestie biznesowe (210)
 - o 8.1.2. Kwestie techniczne (212)
- 8.2. Chmura publiczna i SLA (218)
 - o 8.2.1. SLA dla Amazon AWS (219)
 - o 8.2.2. SLA dla Microsoft Azure (220)
 - o 8.2.3. SLA dla chmury Rackspace (221)
- 8.3. Pomiary jakości operacji w chmurze (222)

- o 8.3.1. Widoczność i monitorowanie u dostawcy jakości świadczonych przez niego usług (222)
- o 8.3.2. Widoczność i monitorowanie jakości usług, które świadczy dostawca, za pomocą rozwiązań zewnętrznych firm (226)
- 8.4. Podsumowanie (228)

9. Przyszłość chmury (229)

- 9.1. Najważniejsza transformacja w dziejach informatyki (231)
 - o 9.1.1. Internet konsumentów oraz chmura (231)
 - o 9.1.2. Chmura w przedsiębiorstwach (235)
- 9.2. Dziesięć prognoz na temat ewolucji chmury (239)
 - o 9.2.1. Tańsza, bardziej niezawodna, bezpieczniejsza i prostsza w użyciu (240)
 - o 9.2.2. Motor napędzający wzrost prekursorów (241)
 - o 9.2.3. Koszty niższe niż w firmowych centrach danych (241)
 - o 9.2.4. Do 2020 roku - 500 tysięcy serwerów wartych miliard dolarów (242)
 - o 9.2.5. Administratorzy a serwery - 1:10 000 w 2020 roku (243)
 - o 9.2.6. Dominacja open source (243)
 - o 9.2.7. Pragmatyczne standardy i rola Amazon API (244)
 - o 9.2.8. Ostateczny standard ISO dla chmury (245)
 - o 9.2.9. Rząd jako prekursor w chmurze (247)
 - o 9.2.10. SaaS i standardy (247)
- 9.3. Dziesięć prognoz na temat ewolucji sposobu wytwarzania aplikacji (248)
 - o 9.3.1. Rola szkieletów aplikacji (248)
 - o 9.3.2. Druga i trzecia warstwa działające w chmurze (249)
 - o 9.3.3. Gwałtowna ewolucja mechanizmów składowania danych (250)
 - o 9.3.4. Lepsza ochrona wrażliwych danych (251)
 - o 9.3.5. Usługi wyższego poziomu o własnych API (252)
 - o 9.3.6. Wzrost znaczenia aplikacji typu mashup (252)
 - o 9.3.7. Dominacja PaaS i FaaS (254)
 - o 9.3.8. Narzędzia do tworzenia aplikacji typu mashup (254)
 - o 9.3.9. Sukces programistów spoza świata zachodniego (255)
 - o 9.3.10. Koszty wytworzenia aplikacji nie są przeszkodą (256)
- 9.4. Podsumowanie (256)
 - o 9.4.1. Pięć podstawowych zasad przetwarzania w chmurze (256)
 - o 9.4.2. Główne zyski z przejścia na chmurę (257)
 - o 9.4.3. Chmura powstała na drodze ewolucji (257)
 - o 9.4.4. Klasyfikacja chmur: od IaaS do SaaS (257)
 - o 9.4.5. Podstawy technologiczne (258)
 - o 9.4.6. Opłaty tylko za rzeczywiste zużycie (258)
 - o 9.4.7. Przesadne obawy o bezpieczeństwo (259)
 - o 9.4.8. Chmury prywatne jako zjawisko tymczasowe (259)
 - o 9.4.9. Projektowanie z myślą o skali i shardowanie (260)
 - o 9.4.10. Projektowanie z myślą o niezawodności i MapReduce (260)
 - o 9.4.11. Lepsze testy, wdrożenia i działanie w chmurze (261)
 - o 9.4.12. Wybór dostawcy (261)
 - o 9.4.13. Monitorowanie chmur publicznych i SLA (261)
 - o 9.4.14. Przyszłość chmury obliczeniowej (261)

A. Powtórka z bezpieczeństwa (263)

- Sekretna komunikacja (264)
- Klucze (265)
- Kryptografia klucza współdzielonego (265)
- Kryptografia klucza publicznego (266)
- XML Signature (268)
- XML Encryption (268)

Skorowidz (271)