

Microsoft Excel 2013: Budowanie modeli danych przy użyciu PowerPivot

Alberto Ferrari i Marco Russo

Przekład:
Marek Włodarz

APN Promise
Warszawa 2014

Microsoft Excel 2013: Budowanie modeli danych przy użyciu PowerPivot

Authorized translation from the English language edition, entitled: MICROSOFT EXCEL 2013: BUILDING DATA MODELS WITH POWERPIVOT (BUSINESS SKILLS), First Edition, ISBN 0735676348 by Alberto Ferrari, published by Pearson Education, Inc, publishing as Microsoft Press, A Division Of Microsoft Corporation.

Copyright © 2013 by Alberto Ferrari and Marco Russo

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Polish language edition published by APN PROMISE S.A., Copyright © 2014

Autoryzowany przekład z wydania w języku angielskim, zatytułowanego: MICROSOFT EXCEL 2013: BUILDING DATA MODELS WITH POWERPIVOT (BUSINESS SKILLS), wydanie pierwsze, SBN 0735676348 by Alberto Ferrari, opublikowanego przez Pearson Education, Inc, publikującego jako Microsoft Press, oddział Microsoft Corporation.

Wszystkie prawa zastrzeżone. Żadna część niniejszej książki nie może być powielana ani rozpowszechniana w jakiegokolwiek formie i w jakikolwiek sposób (elektroniczny, mechaniczny), włącznie z fotokopiowaniem, nagrywaniem na taśmy lub przy użyciu innych systemów bez pisemnej zgody wydawcy.

APN PROMISE SA, ul. Kryniczna 2, 03-934 Warszawa
tel. +48 22 35 51 600, fax +48 22 35 51 699
e-mail: mspress@promise.pl

Książka ta przedstawia poglądy i opinie autorów. Przykłady firm, produktów, osób i wydarzeń opisane w niniejszej książce są fikcyjne i nie odnoszą się do żadnych konkretnych firm, produktów, osób i wydarzeń, chyba że zostanie jednoznacznie stwierdzone, że jest inaczej. Ewentualne podobieństwo do jakiegokolwiek rzeczywistej firmy, organizacji, produktu, nazwy domeny, adresu poczty elektronicznej, logo, osoby, miejsca lub zdarzenia jest przypadkowe i niezamierzone.

Microsoft oraz znaki towarowe wymienione na stronie <http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx> są zastrzeżonymi znakami towarowymi grupy Microsoft. Wszystkie inne znaki towarowe mogą być własnością ich odnośnych właścicieli.

APN PROMISE SA dołożyła wszelkich starań, aby zapewnić najwyższą jakość tej publikacji. Jednakże nikomu nie udziela się rękojmi ani gwarancji.

APN PROMISE SA nie jest w żadnym wypadku odpowiedzialna za jakiegokolwiek szkody będące następstwem korzystania z informacji zawartych w niniejszej publikacji, nawet jeśli APN PROMISE została powiadomiona o możliwości wystąpienia szkód.

ISBN: 978-83-7541-150-8

Przekład: Marek Włodarz
Korekta: Ewa Swędrowska
Skład i łamanie: MAWart Marek Włodarz

Spis treści

<i>Wprowadzenie</i>	ix
1 Wprowadzenie do PowerPivot	1
Korzystanie z tabel przestawnych opartych na tabeli Excela.	2
Korzystanie z PowerPivot w Microsoft Office 2013.	6
Dodawanie informacji do tabeli programu Excel	8
Tworzenie modelu danych zawierającego wiele tabel	10
Istota relacji	12
Istota modelu danych	14
Odpytywanie modelu danych.	16
Dodatek PowerPivot.	17
Korzystanie z narzędzi OLAP i konwertowanie na formuły.	21
Istota PowerPivot dla Excel 2013	26
Tworzenie raportu Power View	29
2 Korzystanie z unikatowych funkcji PowerPivot	37
Ładowanie danych ze źródeł zewnętrznych.	37
Tworzenie tabeli przestawnej PowerPivot.	42
Korzystanie z języka DAX	43
Tworzenie obliczanej kolumny.	44
Tworzenie obliczanego pola	48
Przetwarzanie złożonych agregacji, takich jak liczba wartości odrębnych ...	51
Odświeżanie modelu danych PowerPivot.	55
3 Wprowadzenie do DAX	57
Istota obliczeń DAX	57
Składnia DAX.	58
Typy danych DAX	59
Operatory DAX	61
Wartości DAX.	62
Kolumny i pola obliczeniowe	62
Kolumny obliczane	63
Pola obliczeniowe	64

Obsługa błędów w wyrażeniach DAX	69
Błędy konwersji	69
Operacje arytmetyczne	70
Przechwytywanie błędów	73
Formatowanie kodu DAX	75
Typowe funkcje DAX	77
Funkcje agregujące	78
Funkcje logiczne	81
Funkcje informacyjne	83
Funkcje matematyczne	84
Funkcje tekstowe	85
Funkcje konwersji	86
Funkcje daty i czasu	86
Funkcje relacyjne	87
Korzystanie z podstawowych funkcji DAX	89
4 Istota modeli danych	93
Podstawy modelowania danych	94
Tworzenie raportu bez użycia modelu danych	95
Budowanie modelu danych	98
Więcej na temat relacji	100
Normalizacja i denormalizacja	102
Denormalizacja wewnątrz zapytań SQL	106
Projektant zapytań PowerPivot	107
Kiedy denormalizować tabele?	115
Nadmierna denormalizacja	116
OLTP i hurtownie danych	119
Odpytywanie bazy danych OLTP	120
Tematyczne hurtownie danych, fakty i wymiary	123
Schematy gwiazdy	125
Która baza danych jest najlepszym źródłem?	126
Korzystanie z zaawansowanych relacji	127
5 Publikowanie w SharePoint	131
Integracja SharePoint 2013 i PowerPivot	131
Licencjonowanie i konfiguracja	132
Publikowanie skrótytu w witrynie SharePoint	132
Korzystanie z Galerii programu PowerPivot	137
Łączenie programu Excel z modelem danych SharePoint	139
Tworzenie raportu Power View	142
Zarządzanie odświeżaniem danych PowerPivot	144

6 Ładowanie danych	149
Istota źródeł danych	149
Ładowanie z bazy danych	152
Ładowanie z listy tabel	154
Ładowanie relacji	157
Wybieranie powiązanych tabel	158
Ładowanie z zapytania SQL	159
Ładowanie z widoków	159
Otwieranie istniejących połączeń	160
Ładowanie z bazy danych programu Access	162
Ładowanie danych z SQL Server Analysis Services	164
Korzystanie z edytora MDX	166
Obsługa kluczy w kostce OLAP	170
Ładowanie z bazy tabelarycznej	171
Ładowanie z SharePoint	173
Korzystanie z połączonych tabel	178
Ładowanie z plików Excela	181
Ładowanie z plików tekstowych	184
Wczytywanie danych ze schowka	186
Ładowanie z raportu	188
Ładowanie ze strumienia danych	193
Ładowanie danych z Windows Azure Marketplace	195
Sugerowane dane pokrewne	199
Odświeżanie połączeń	199
7 Istota kontekstów wykonania	203
Wprowadzenie do kontekstów wykonania	204
Pojęcie kontekstu wiersza	208
Testowanie zrozumienia kontekstów wykonania	210
Użycie funkcji <i>SUM</i> w kolumnie obliczanej	210
Użycie pól w polu obliczeniowym	211
Tworzenie kontekstu wiersza poprzez iteratory	213
Iteratory <i>FILTER</i> , <i>ALL</i> i interakcje kontekstów	217
Praca z wieloma tabelami	220
Kontekst wiersza a relacje	221
Kontekst filtru a relacje	224
Funkcja <i>VALUES</i>	225
Funkcje <i>ISFILTERED</i> oraz <i>ISCROSSFILTERED</i>	226
Podsumowanie kontekstów wykonania	230
Tworzenie tabeli parametrów	232
8 Funkcja <i>CALCULATE</i>	235
Dlaczego potrzebna jest funkcja <i>CALCULATE</i> ?	235

Przykłady zastosowań <i>CALCULATE</i>	240
Filtrowanie pojedynczej kolumny.	240
Filtrowanie przy użyciu złożonych warunków	244
Używanie funkcji <i>CALCULATE</i> wewnątrz kontekstu wiersza	249
Zależności cykliczne	252
Reguły dotyczące <i>CALCULATE</i>	257
Funkcja <i>ALLSELECTED</i>	258
9 Korzystanie z hierarchii	261
Istota hierarchii	261
Kiedy budować hierarchie	263
Budowanie hierarchii	264
Tworzenie hierarchii opartych na wielu tabelach	265
Wykonywanie obliczeń przy użyciu hierarchii	267
Używanie hierarchii drzewiastych	278
10 Korzystanie z Power View	291
Czym jest Power View?	291
Podstawy Power View	292
Korzystanie z panelu Filtry	296
Dekorowanie raportu	298
Tabele, macierze i karty	299
Korzystanie z wizualizacji macierzowej	299
Wizualizacja kart	301
Używanie tabeli jako fragmentatora	303
Stosowanie wykresów	304
Korzystanie z wykresu liniowego	305
Korzystanie z wykresów kołowych	305
Korzystanie z wykresów punktowych	307
Korzystanie z map	309
Drażnienie w głąb	313
Korzystanie z kafelków	315
Wielokrotności	318
Efektywne wykorzystanie Power View	320
11 Kształtowanie raportów	321
Wskaźniki KPI	321
Tworzenie modeli danych na potrzeby Power View	333
Metadane Power View	338
Sumowanie kolumn	340
Korzystanie z domyślnego zestawu pól	342
Korzystanie z okna dialogowego Zachowanie tabeli	343
Definiowanie zestawów	345

Tworzenie dynamicznych zestawów przy użyciu MDX.	350
Korzystanie z perspektyw	355
Drażenie danych.	359
12 Wykonywanie obliczeń na danych w języku DAX	361
Budowanie tabeli kalendarzowej.	361
Korzystanie z wielu tabel kalendarza.	365
Obliczanie dni roboczych	368
Wyliczanie różnicy w dniach roboczych	375
Agregowanie i porównywanie danych względem czasu	380
Od początku roku (YTD), kwartału (QTD) i miesiąca (MTD).	380
Analiza czasowa wykorzystująca <i>CALCULATE</i>	384
Obliczenia dla roku poprzedniego (<i>prior year</i> – PY).	387
Obliczanie rocznej sumy ruchomej	390
Korzystanie z innych funkcji agregujących.	392
Wyliczanie różnicy względem roku poprzedniego	393
Zamknięcie salda względem czasu	395
Miary częściowo agregowalne	395
Funkcje <i>OPENINGBALANCE</i> i <i>CLOSINGBALANCE</i>	399
Aktualizowanie sald przy użyciu transakcji.	403
Obliczanie średnich ruchomych.	406
13 Zaawansowane wyrażenia DAX	413
Podział na przedziały	413
Pozycjonowanie	419
Relacje wiele-do-wielu.	427
Wyliczanie nowych i powracających klientów	432
Funkcja <i>KEEPFILTERS</i>	437
Implementacja analizy koszyka	447
Potęga kolumn obliczeniowych: Analiza ABC	455
Obsługa konwersji walut.	459
14 DAX jako język zapytań.	465
Funkcja <i>EVALUATE</i>	466
Tworzenie tabeli Excela za pomocą <i>EVALUATE</i>	467
Używanie typowych funkcji w zapytaniach	472
Funkcja <i>FILTER</i>	472
Funkcja <i>CALCULATETABLE</i>	473
Korzystanie z funkcji <i>ADDCOLUMNS</i>	474
Korzystanie z funkcji <i>VALUES</i> w połączeniu z <i>ADDCOLUMNS</i>	476
Korzystanie z funkcji <i>SUMMARIZE</i>	477
Korzystanie z opcji <i>ROLLUP</i>	480
Łączenie zwrotne zapytania DAX.	481

Przetwarzanie analizy ABC przy użyciu tabeli połączonej zwrotnie.	485
Funkcja <i>CROSSJOIN</i>	489
Funkcja <i>GENERATE</i>	490
Zapytania przy użyciu DAX Studio.	493
15 Automatyzowanie operacji przy użyciu VBA	495
Włączanie karty DEWELOPER we wstążce	495
Aktualizowanie zwrotnie połączonego zapytania DAX poprzez VBA	501
Korzystanie z obiektu <i>Model</i>	506
Importowanie danych do modelu przy użyciu VBA	507
Istota połączeń danych	511
16 Porównanie PowerPivot for Excel i SQL Server Analysis Services.	515
Rozróżnianie poszczególnych wersji mechanizmu	516
Zestaw funkcji	516
Zabezpieczanie danych	518
Programowalność i elastyczność	520
Tłumaczenia.	523
Wielkość bazy danych	523
Liczba baz danych	526
PowerPivot jako system prototypowy.	527
A Nazwy funkcji	529
Indeks.	539
O autorach	557

Wprowadzenie

Microsoft Excel jest niekwestionowanym, światowym standardem w dziedzinie wykonywania analizy danych. Łatwość użycia i bogactwo możliwości sprawiają, że arkusze Excela są narzędziem używanym przez wszystkich, niezależnie od rodzaju analizowanych informacji.

Możemy wykorzystywać Excela do rejestrowania osobistych wydatków, bieżącego stanu konta, danych klientów lub złożonego planu biznesowego, a nawet postępów w utracie wagi podczas trudnej do utrzymania diety. Możliwości są nieograniczone – nie zamierzamy nawet zaczynać wyliczać wszystkich rodzajów informacji, które można analizować w Excelu. Faktem jest, że jeśli mamy jakieś dane, które można uporządkować i przeanalizować, istnieje wielkie prawdopodobieństwo, że Excel będzie doskonałym narzędziem do tego celu. Można w nim łatwo uporządkować dane w formie tabel, aktualizować je, tworzyć wykresy, tabele przestawne i obliczenia oparte na nich, a następnie tworzyć prognozy – wszystko przy stosunkowo niewielkiej znajomości oprogramowania. Wraz z pojawieniem się chmury można obecnie używać Excela nawet na urządzeniach mobilnych, takich jak tablety i smartfony, wykorzystując Internet w celu utrzymania ciągłego dostępu do informacji. Zauważmy też, że we wcześniejszych wersjach Excel istniał limit 65536 wierszy dla jednego arkusza, ale fakt, że tak wielu klientów zwracało uwagę na potrzebę powiększenia tej wartości (co firma Microsoft zrobiła, podnosząc tę granicę do 1 miliona wierszy w wersji Excel 2007), jest jasnym wskazaniem, że użytkownicy chcieli mieć możliwość przechowywania i analizowania wielkich ilości danych.

Oprócz użytkowników Excela istnieje jeszcze druga kategoria ludzi, dedykujących swoje życie zawodowe analizom danych: są to profesjonaliści analityki biznesowej (*business intelligence* – BI). BI jest nauką o wyciąganiu wniosków i spostrzeżeń z wielkich zbiorów danych i w minionych latach profesjonaliści z tej dziedziny opanowali i stworzyli wiele nowych technik i narzędzi do zarządzania systemami, które są w stanie obsłużyć zbiory danych wielkości setek milionów, a może nawet miliardów wierszy. Systemy BI wymagają jednak pracy wielu specjalistów i kosztownego, wydajnego sprzętu. Są bardzo potężne, ale kosztowne, a ich budowanie trwa długo, co stanowi poważne wady.

Przed rokiem 2010 istniało jasne rozgraniczenie pomiędzy analizami na poziomie małych i wielkich ilości danych: po jednej stronie był Excel, a po drugiej rozbudowane systemy BI. Pierwszy krok w kierunku scalenia, a przynajmniej zbliżenia obydwu światów, został zrobiony już dawno, gdyż narzędzie tabel przestawnych w Excelu miało

możliwość odpytywania systemów BI. W ten sposób analitycy mogli wykonywać zapytania do wielkich systemów BI i uzyskać najlepsze cechy obu światów, gdyż rezultaty takich zapytań mogli umieszczać w tabelach przestawnych Excela i następnie używać ich do wykonywania dalszych analiz.

W roku 2010 firma Microsoft wykonała znaczący ruch w kierunku obalenia muru oddzielającego profesjonalistów BI od użytkowników Excela, wprowadzając xVelocity, wydajny mechanizm bazodanowy, który pozwala tworzyć spore rozwiązania BI bezpośrednio w Excelu. Wydarzeniem tym było opublikowanie Microsoft SQL Server 2008 R2 PowerPivot for Excel jako bezpłatnego dodatku dla programu Excel 2010. Celem było takie ułatwienie kreacji rozwiązań BI, by Excel stał się nie tylko klientem BI, ale również serwerem, zdolnym do utrzymywania złożonych rozwiązań BI. Koncepcji tej nadano nazwę „samoobsługowych rozwiązań BI” (*self-service BI*).

Microsoft PowerPivot nie ma sztywnego ograniczenia co do liczby wierszy tabeli, którą może przechować: jeśli zachodzi potrzeba obsłużenia stu milionów wierszy, można to zrobić, a prędkość analizy nadal będzie zadziwiająca. Wraz z PowerPivot pojawił się również język DAX, rozbudowany język programowania ukierunkowany na tworzenie rozwiązań BI, a nie tylko formuł Excela. Na koniec PowerPivot jest w stanie kompresować dane w takim stopniu, że ogromne ilości danych mogą być przechowywane w relatywnie niedużych plikach skoroszytów. Jednak to był tylko pierwszy krok.

Drugim, zdecydowanym ruchem w kierunku dostarczenia siły analityki biznesowej samym użytkownikom było wprowadzenie programu Excel 2013. PowerPivot nie jest już oddzielnym dodatkiem do Excela; obecnie jest to nieodłączny element technologii Excela, udostępniający siłę mechanizmu xVelocity każdemu użytkownikowi. Tak więc era samoobsługowych rozwiązań BI zaczęła się w roku 2010, a od 2013 przeszła już w stan zaawansowany.

Ktoś, kto czyta to wprowadzenie, zapewne jest zainteresowany dołączeniem do świata samoobsługowych rozwiązań BI i chce nauczyć się, jak posługiwać się i udoskonalać swoje rozwiązania w PowerPivot for Excel. Musi opanować podstawy narzędzia, ale to tylko początek. Później konieczne będzie nauczanie się, jak kształtować dane, aby móc wydajnie wykonywać analizy: nazywamy to *modelowaniem danych*. Na koniec trzeba będzie nauczyć się języka DAX i opanować jego wszystkie koncepcje, aby móc osiągnąć najlepsze efekty przy jego użyciu. Jeśli to właśnie interesuje Czytelnika, to znalazł odpowiednią książkę.

My, autorzy tej książki, jesteśmy zawodowcami w dziedzinie BI i wiemy z doświadczenia, że budowanie rozwiązań BI nie jest łatwe. Nie chcielibyśmy tworzyć złudnych nadziei: BI to fascynujące zagadnienie, ale również bardzo trudne. Książka ma pomóc Czytelnikowi w podjęciu koniecznych działań, aby przekształcić się ze „zwykłego” (choć zapewne zaawansowanego) użytkownika Excela w projektanta samoobsługowych rozwiązań BI. To będzie długa droga, wymagająca czasu i determinacji, aby móc naprawdę opanować nowe techniki. Jednak rezultaty, które w ten sposób będzie można osiągnąć, są nieocenione.

Książka ta nie jest podręcznikiem typu krok po kroku. Jeśli ktoś szuka pozycji typu PowerPivot dla opornych, to nie jest książka dla niego. Jeśli jednak ktoś szuka książki, która zamiast pokazywać gotowe rozwiązania, wskaże kierunek, w którym należy podążać i zabierze go w długą, ale ciekawą podróż, witamy na pokładzie!

Podczas pisania tej książki postanowiliśmy skoncentrować się na koncepcjach i przykładach z realnego świata, zaczynając od zera i pokazując drogę w kierunku osiągnięcia mistrzostwa w posługiwaniu się językiem DAX. Nie omówiliśmy wszystkich poszczególnych funkcji ani nie wyjaśniamy działań w stylu „kliknij to, a potem tamto”. Z drugiej strony, upakowaliśmy w tej książce prawdziwą masę informacji, dzięki którym, po skończeniu studiowania tej pozycji, Czytelnik będzie miał solidne podstawy w dziedzinie modelowania danych w Excelu.

Ostatnie zdanie podkreśla główną cechę tej książki: to książka do studiowania, a nie tylko do przeczytania. Przed nami długa wycieczka, ale obiecujemy, że nikt nie będzie żałował poświęconego czasu i wysiłku.

UWAGA Funkcje PowerPivot oraz Power View są zawarte tylko w wybranych konfiguracjach Microsoft Office 2013. Funkcja PowerPivot, która była dostępna jako dodatek dla wszystkich wydań Microsoft Excel 2010, jest dostępna tylko w wydaniach Office 2013 Professional Plus, SharePoint 2013 Enterprise Edition, SharePoint Online 2013 Plan 2 oraz wydaniach E3 i E4 pakietu Office 365. Funkcja Power View, stanowiąca nowy element programu Excel 2013, zawarta jest w tych samych wersjach co PowerPivot. Szczęśliwie *Excel Data Model* jest obsługiwany we wszystkich konfiguracjach programu Excel 2013. Trzeba jednak mieć na uwadze, że dostępność konfiguracji może się zmienić.



Dla kogo jest ta książka

Książka adresowana jest do użytkowników programu Excel, menedżerów projektów i osób decyzyjnych, którzy chcą poznać podstawy PowerPivot for Excel 2013, opanować nowy język DAX używany przez PowerPivot i poznać zaawansowane modelowanie danych oraz techniki programowania wykorzystywane w PowerPivot.

Założenia

Pisząc tę książkę założyliśmy, że Czytelnik ma opanowane podstawy programów Excel 2010 lub Excel 2013. Nie musi być mistrzem, ale przynajmniej częstym użytkownikiem. Omówimy informacje potrzebne do przejścia z posługiwania się samym programem Excel na korzystanie z PowerPivot, ale nie będziemy tracić czasu ani miejsca na podstawy Excela, takie jak wprowadzanie formuł, konstruowanie funkcji takich jak WYSZUKAJ.PIONOWO (VLOOKUP) czy inne podstawowe funkcjonalności.

Nie jest wymagana żadna wcześniejsza znajomość PowerPivot. Jeśli ktoś próbował już samodzielnie zbudować model danych, to świetnie, ale zakładamy, że Czytelnik przed rozpoczęciem lektury nigdy nie widział jeszcze okna PowerPivot.

Organizacja książki

Książka została zaprojektowana z myślą o lekturze „liniowej” – od deski do deski. Próba przeskakiwania bezpośrednio do rozwiązania określonego problemu, z pominięciem jakiegoś fragmentu, będzie zapewne złym wyborem. W każdym rozdziale przedstawimy koncepcje i funkcjonalności, które będą potrzebne do zrozumienia kolejnych rozdziałów. Co więcej, napisaliśmy niektóre rozdziały ze świadomością, że Czytelnik będzie musiał do nich powrócić (być może nie raz), gdyż teoretyczne podstawy w nich zawarte są trudne do opanowania w pierwszym czytaniu.

Książka podzielona jest na 16 rozdziałów:

Rozdział 1, „Wprowadzenie do PowerPivot”, proponuje „wycieczkę z przewodnikiem” po podstawowych funkcjach PowerPivot for Excel 2013. Pokażemy w nim główne korzyści z wykorzystania PowerPivot do zaspokojenia potrzeb analitycznych. W rozdziale tym również pokażemy tworzenie prostego raportu Power View.

Rozdział 2, „Korzystanie z unikatowych funkcji PowerPivot”, prezentuje funkcjonalności, które są dostępne dopiero po uaktywnieniu dodatku PowerPivot for Excel. Należą do nich kolumny obliczane, miary (pola obliczeniowe), hierarchie i kilka innych podstawowych funkcjonalności. Rozdział ten jest logiczną kontynuacją (i dopełnieniem) rozdziału 1.

W rozdziale 3, „Wprowadzenie do DAX”, zaczniemy omawiać język DAX, pokazując jego składnię i najbardziej podstawowe funkcje. Podkreślimy różnice pomiędzy kolumną obliczaną a polem obliczeniowym, a na koniec pokażemy pierwszy praktyczny przykład zastosowania języka DAX.

Rozdział 4, „Istota modeli danych”, to teoretyczne wprowadzenie do modelowania danych, pokazujące różne opcje modelowania dostępne w bazie danych PowerPivot. Opiszemy kilka koncepcji, które nie muszą być oczywiste dla użytkowników Excel, takich jak normalizacja i denormalizacja, budowa zapytań SQL, jak działają relacje i dlaczego są takie ważne, a także struktury hurtowni danych.

W rozdziale 5, „Publikowanie w SharePoint”, omówimy proces publikowania skoroszytów na serwerze Microsoft SharePoint, aby umożliwić pracę zespołową. Ponadto pokażemy ideę PowerPivot for SharePoint, który jest aplikacją serwerową, którą można kontrolować i rozbudowywać używając Excela i PowerPivot.

Rozdział 6, „Ładowanie danych”, poświęcony jest różnym sposobom ładowania danych do bazy PowerPivot. Dla każdego źródła danych pokażemy, jak działa i proponujemy wiele wskazówek oraz najlepszych praktyk dla poszczególnych źródeł.

W rozdziale 7, „Istota kontekstów wykonania” oraz 8, „Funkcja CALCULATE” powrócimy do teorii – jest to prawdziwy rdzeń książki. Pokażemy tu koncepcje

kontekstów wykonania, relacji oraz działanie funkcji *CALCULATE*. Są to filary języka DAX i konieczne jest ich doskonale opanowanie, zanim będzie możliwe budowanie zaawansowanych modeli danych w PowerPivot.

Rozdział 9, „Korzystanie z hierarchii”, pokazuje, jak tworzyć i zarządzać hierarchiami. Przedstawimy tu podstawy hierarchii, obliczanie wartości przy ich użyciu, a na koniec pokażemy, jak obsłużyć hierarchie drzewiaste, wykorzystując koncepcje poznane w rozdziałach 7 i 8.

Rozdział 10, „Korzystanie z Power View”, poświęcony jest nowemu narzędziu raportowania dostępnemu w programie Excel 2013: Power View. Pokażemy główne funkcje tego narzędzia, tworzenie prostych raportów Power View oraz jak filtrować dane i budować raporty, które zarazem wyglądają przyjemnie i zapewniają użyteczny wgląd w nasze dane.

Rozdział 11, „Kształtowanie raportów”, omawia wiele zaawansowanych zagadnień dotyczących raportowania. Omówimy tu wskaźniki KPI (Key Performance Indicator), ich tworzenie i stosowanie w celu podniesienia jakości systemu raportowania. Pokażemy tu również warstwę metadanych Power View w PowerPivot, drążenie w głąb, budowanie zbiorów poprzez interfejs Excela lub język MDX, a także perspektywy.

Rozdział 12, „Wykonywanie obliczeń na danych w języku DAX”, zajmuje się analizami czasowymi (*time intelligence*). Obliczenia wartości YTD (Year to Date – od początku roku) i pochodnych, wyliczanie dni roboczych dla zadanego okresu, miary semiad-dytywne, średnie ruchome i inne skomplikowane kalkulacje wykorzystujące czas są omówione w tym rozdziale.

Rozdział 13, „Zaawansowane wyrażenia DAX”, to kolekcja scenariuszy i rozwiązań, które wszystkie mają tę samą cechę: trudno je rozwiązać przy użyciu samego Excela czy innego narzędzia, podczas gdy okazują się łatwiejsze do opanowania w języku DAX, gdy już uzyskaliśmy niezbędną wiedzę z poprzednich rozdziałów książki. Wszystkie te przykłady pochodzą z rzeczywistych sytuacji i należą do najliczniejszych zapytań, z którymi mieliśmy do czynienia jako konsultanci lub widywanych na tematycznych forum w sieci Web.

Rozdział 14, „DAX jako język zapytań”, zgodnie z tytułem poświęcony jest wykorzystaniu DAX jako języka zapytań. Obejmuje różne funkcjonalności, które pozwalają użyć DAX do odpytywania bazy danych PowerPivot. Pokażemy tu również bardziej zaawansowane rozwiązania, jak odwrotnie połączone lub zwrotnie połączone tabele, które znacznie rozszerzają możliwości budowania złożonych modeli danych PowerPivot.

Rozdział 15, „Automatyzowanie operacji przy użyciu VBA”, omawia wykorzystanie języka Microsoft Visual Basic for Applications (VBA) do zarządzania skoroszytami PowerPivot w sposób programowy i automatyzowania pewnych typowych zadań. Zamieściliśmy w nim kilka przykładów kodu i pokazaliśmy, jak rozwiązać niektóre najbardziej typowe scenariusze, w których VBA okazuje się użyteczny.

Rozdział 16, „Porównanie Excela i SQL Server Analysis Services”, zawiera zestawienie funkcjonalności trzech odmian technologii PowerPivot: PowerPivot for Excel, PowerPivot for SharePoint oraz SQL Server Analysis Services (SSAS). Celem tego końcowego rozdziału jest zapewnienie jasnego obrazu, co można zrobić w PowerPivot for Excel, kiedy trzeba wykonać krok dalej i zaadaptować PowerPivot for SharePoint, i jakie dodatkowe funkcjonalności zapewnia jedynie SSAS.

Konwencje

W książce tej użyte zostały następujące konwencje:

- **Druk wytłuszczony** oznacza tekst, który należy wpisać.
- *Kursywą* oznaczone są nowe terminy, nazwy pól obliczeniowych i kolumn oraz nazwy baz danych.
- Nazwy poleceń, opcji menu, okien dialogowych i innych elementów interfejsu zostały wyróżnione **półgrubą czcionką bezszeryfową**.
- Nazwy funkcji (słów kluczowych) języka DAX, języka formuł Excela i języka SQL oraz MDX pisane są *WIELKIMI LITERAMI*.
- Skróty klawiszowe są oznaczone poprzez użycie znaku plus (+) pomiędzy nazwami klawiszy. Na przykład Ctrl+Alt+Delete oznacza, że należy nacisnąć klawisze Ctrl, Alt i Delete w tym samym czasie.

Zawartość towarzysząca książce

Treść towarzysząca książce, która ułatwi naukę i wzbogaci doświadczenia Czytelnika, można pobrać z następującej strony Web (w zakładce **Downloads**):

<http://go.microsoft.com/fwlink/?Linkid=279953>

Treści te obejmują następujące elementy:

- Wersję Microsoft Access bazy danych *AdventureWorksDW*, której można użyć do samodzielnego zbudowania wszystkich przykładów omawianych w książce.
- Wszystkie skoroszyty Excela wzmiankowane w tekście (czyli wszystkie skoroszyty, które zostały użyte do zilustrowania przykładów i koncepcji). Do otwarcia tych plików niezbędny jest program Excel 2013.

Podziękowania

Powinniśmy podziękować tak wielu ludziom za ich pomoc w pisaniu książki, że niemożliwe jest wypisanie pełnej listy. Zatem dziękujemy wszystkim, którzy przyczynili się do jej powstania – nawet jeśli nie wiedzą oni, że to zrobili. Komentarze na blogach, posty w grupach dyskusyjnych, dyskusje prze email, rozmowy z uczestnikami i wykładowcami na konferencjach technicznych i wiele więcej – wszystko to było użyteczne i potrzebne, a wielu z tych ludzi wniosło znaczące idee do tej książki. Są jednak osoby, które chcemy wymienić osobiście ze względu na ich szczególny udział.

Na początek Edward Melomed: zainspirował nas i zapewne nie zaczęlibyśmy naszej przygody z PowerPivot bez zawziętej dyskusji, którą toczyliśmy z nim kilka lat temu.

Chcemy podziękować Microsoft Press, O'Reilly Media i ludziom stamtąd, którzy przyczynili się do realizacji tego projektu: Kenyonowi Brownowi, Christopherowi Hearse i wielu innym, którzy pracowali poza światłami rampy.

Jedynym zadaniem dłuższym od pisania książki są studia, które trzeba przeprowadzić, aby móc ją napisać. Grupa ludzi, którą (przyjacielsko) nazywamy „ssas-insiders”, pomogła nam przygotować się do pisania tej książki. Kilka osób z firmy Microsoft zasługuje na szczególne wyróżnienie, gdyż poświęcili nam sporo ich cennego czasu, ucząc nas ważnych koncepcji PowerPivot i DAX. Ich nazwiska to Marius Dumitru, Jeffrey Wang oraz Akshai Mirchandani. Wasza pomoc była bezcenna!

Chcemy też podziękować Amirowi Netz, Ashvini Sharma oraz T. K. Anand za ich udział w dyskusji, jak spozycjonować PowerPivot.

Dokończenie książki w czasach Internetu jest trudne, gdyż ciągle napływają nowe pomysły i idee. Kilka blogów było szczególnie ważnych dla naszej książki i chcemy wymienić tu ich twórców: Chris Webb, Kasper de Jonge, Rob Collie, Denny Lee oraz Dave Wickert.

Na koniec specjalne podziękowania należą się recenzentowi technicznemu, Javierowi Guillen. Dwukrotnie sprawdził całą zawartość oryginalnego tekstu, wyszukując błędy i dając nam nieocenione sugestie, jak można poprawić tę książkę. Jeśli finalny tekst zawiera mniej błędów, niż oryginalny manuskrypt, to tylko dzięki niemu. Jeśli zaś nadal są tu jakieś błędy, to jest to nasza wina, oczywiście.

Dziękujemy jeszcze raz!

Errata do książki i pomoc

Dołożyliśmy wszelkich starań, aby zapewnić najwyższą jakość tej książki i towarzyszącym jej materiałom. W przypadku wykrycia błędów proszę zgłosić je na stronie Microsoft Press w Oreilly.com:

1. Wejść na stronę <http://microsoftpress.oreilly.com>.
2. W polu **Search** wpisać numer ISBN lub tytuł książki (oryginalnego, angielskiego wydania – informacje te można znaleźć na stronie redakcyjnej po stronie tytułowej).
3. Zaznaczyć książkę w wynikach wyszukiwania.
4. Na stronie katalogowej książki, pod zdjęciem okładki, odszukać listę łączy.
5. Kliknąć **View/Submit Errata**.

Dodatkowe informacje i usługi dotyczące tej książki można znaleźć na jej stronie katalogowej. Aby uzyskać dodatkową pomoc, proszę pisać do działu Microsoft Press book Support na adres mspinput@microsoft.com.

Proszę pamiętać, że podane wyżej adresy nie oferują pomocy na temat produktów firmy Microsoft.

Prośba o uwagi

Najwyższym priorytetem dla wydawnictwa Microsoft Press jest satysfakcja Czytelnika i dlatego wszelkie uwagi są bardzo cenne. Prosimy o dzielenie się z nami swoimi opiniami na stronie:

<http://www.microsoft.com/learning/booksurvey>.

Czytamy wszystkie komentarze i uwagi. Z góry dziękujemy za odzew!

Bądźmy w kontakcie

Kontynuujmy dyskusję! Jesteśmy na Tweeterze: <http://twitter.com/MicrosoftPress>.

ROZDZIAŁ 1

Wprowadzenie do PowerPivot

Microsoft PowerPivot dla Microsoft Excel 2013 to rozwiązanie zorientowane na dostarczanie samoobsługowych rozwiązań typu *business intelligence* (BI), stanowiące prawdziwą rewolucję w świecie analizy danych, gdyż daje ono końcowemu użytkownikowi pełne możliwości i siłę potrzebne do wykonywania złożonych analiz bez konieczności angażowania specjalistów BI i techników. PowerPivot stanowi rozszerzenie (*add-in*) programu Excel, które implementuje szybką, wydajną, zlokalizowaną w pamięci bazę danych, którą można wykorzystać do porządkowania danych, wykrywania interesujących zależności i zapewnienia najszybszych sposobów przeglądania informacji.

Do najbardziej interesujących funkcjonalności PowerPivot należą:

- Zdolność do zorganizowania tabel danych dla raportów tabel i wykresów przedstawianych w sposób relacyjny, co uwalnia analityka od konieczności wstępnego zaimportowania danych jako arkuszy programu Excel przed przystąpieniem do ich analizowania.
- Dostępność szybkiej, oszczędnej pamięciowo kolumnowej bazy danych, która pozwala pomieścić ogromne ilości danych bez ograniczeń charakterystycznych dla arkuszy programu Excel.
- DAX, czyli wydajny język programowania umożliwiający definiowanie złożonych wyrażeń odnoszących się do relacyjnej bazy danych. Umożliwia on tworzenie zaskakująco bogatych wyrażeń, gdy porównamy je ze standardowymi funkcjami Excela.
- Zdolność do integrowania danych pochodzących z rozmaitych źródeł, takich jak bazy danych, arkusze Excela czy źródła dostępne w Internecie – w ostatecznym rachunku dane dowolnego rodzaju.
- Zadziwiająco szybkie przetwarzanie złożonych zapytań wobec całej bazy danych.

Niektórzy użytkownicy mogą myśleć o PowerPivot jako o prostym następcy od dawnego znanego narzędzia PivotTable, podczas gdy inni będą go używać jako narzędzia szybkiego projektowania złożonych rozwiązań BI. Jeszcze inni mogą sądzić, że jest to prawdziwy zamiennik dla złożonych rozwiązań BI. W rzeczywistości PowerPivot nie jest w stanie zastąpić wielkich i złożonych rozwiązań BI, takich jak te oparte na mechanizmach Microsoft Analysis Services, ale jest czymś znacznie więcej, niż

prostim zamiennikiem dla PivotTable i stanowi wspaniałe narzędzie poznawania świata BI i implementowania rozwiązań BI.

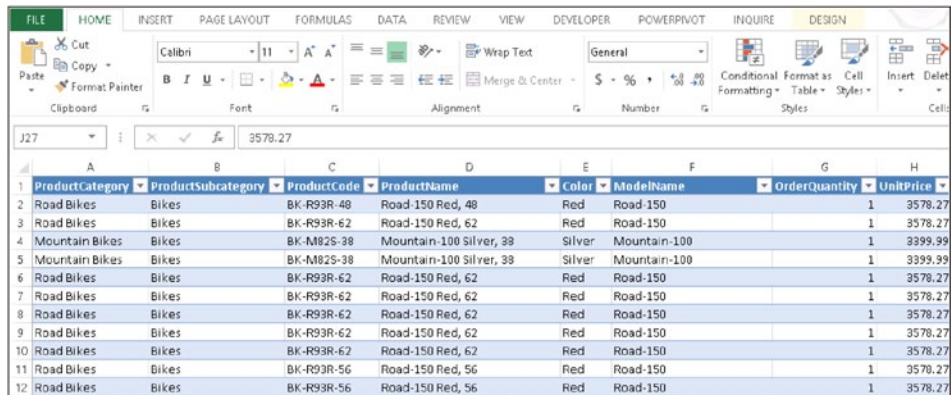
PowerPivot wypełnia lukę pomiędzy arkuszem Excela a pełnym rozwiązaniem BI i ma kilka unikatowych cech, które sprawiają, że jest atrakcyjnym narzędziem zarówno dla zaawansowanych użytkowników programu Excel, jak i dla doświadczonych analityków BI. W tej książce przedstawimy wszystkie funkcje i cechy PowerPivot, jednak podobnie jak przy każdym wielkim projekcie, musimy rozpocząć od podstaw. Rozdział ten rozpoczyna się prostym wprowadzeniem do podstawowych funkcjonalności PowerPivot. Sugerujemy, by Czytelnik na bieżąco wykonywał przedstawiane instrukcje, aby mógł zobaczyć na ekranie własnego komputera te rezultaty, które pokazujemy w książce. Później, w kolejnych rozdziałach, nie będziemy już używali tego typu instrukcji „krok po kroku”, gdyż uważamy, że lepiej wykorzystamy miejsce na stronach książki, skupiając się na koncepcjach i ideach, niż marnować je na polecenia w rodzaju „kliknij Next”.

Choć książka ta poświęcona jest narzędziu PowerPivot dla Excel 2013, dobrym pomysłem wydaje się rozpoczęcie od krótkiego przypomnienia, jak narodziła się idea tego rozwiązania i jak ono działało w wersji Excel 2010, dzięki czemu łatwiej będzie można docenić siłę nowych funkcjonalności i zrozumieć niektóre szczególne cechy tego rozszerzenia.

Korzystanie z tabel przestawnych opartych na tabeli Excela

Rozpoczniemy od krótkiej wycieczki w przeszłość. Już od czasu opublikowania wersji Excel 97 możliwe było wykonywanie pewnych analiz danych przy użyciu tabel przestawnych, czyli narzędzia PivotTable. W istocie, do czasu pojawienia się rozszerzenia PowerPivot tabele przestawne były główną metodą analizowania danych. Stanowią one łatwą i wygodną metodę przeglądania wielkich ilości danych zgromadzonych w arkuszach programu Excel. W książce tej nie będziemy zagłębiać się w szczegóły działania tego narzędzia; istnieje wystarczająco wiele dobrych wyjaśnień i wskazówek w innych źródłach. Tym niemniej przydatne będzie przypomnienie podstawowych funkcjonalności tabel przestawnych, aby móc porównać je z tymi, które udostępnia PowerPivot.

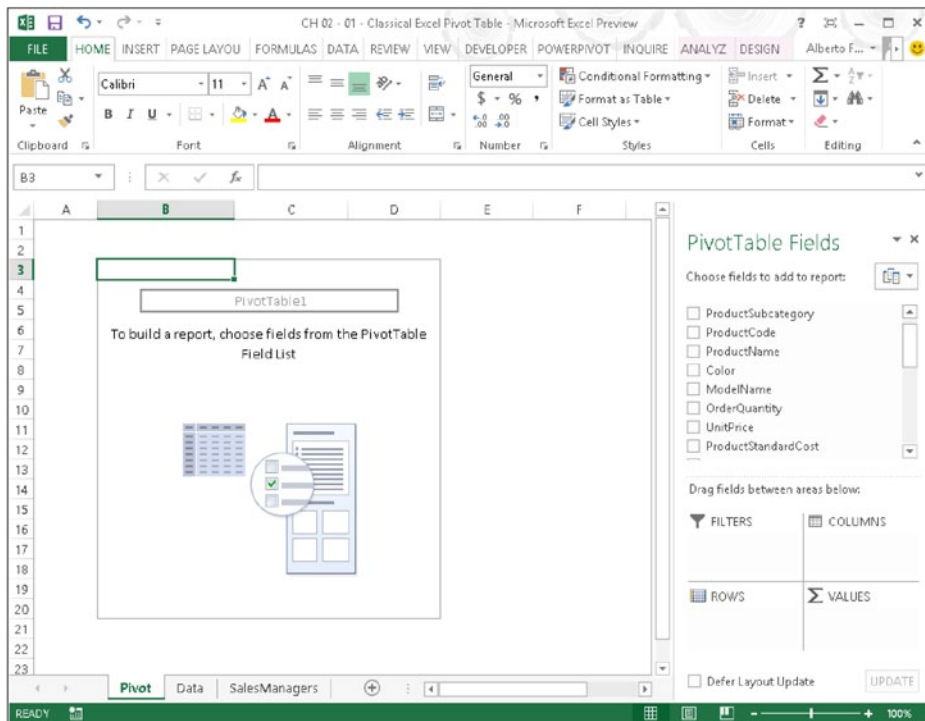
Przyjmijmy, że mamy standardową tabelę programu Excel, zaimportowaną poprzez kwerendę wykonaną w bazie danych, zawierającą wszystkie dane, które chcielibyśmy przeanalizować. Aby uzyskać takie dane, zwykle trzeba zwrócić się do działu IT o udostępnienie pewnych narzędzi dostępu do bazy danych i sformułowanie kwerendy wydobywającej potrzebne informacje. Wynikowy arkusz może wyglądać podobnie do tego pokazanego na rysunku 1-1. Ponieważ tabela ta zawiera surowe dane, są one trudne do zinterpretowania i analizy. Pokazany arkusz zawarty jest w plikach powiązanych z książką jako *CH01-01-Classical Excel PivotTable.xlsx*.



	A	B	C	D	E	F	G	H
1	ProductCategory	ProductSubcategory	ProductCode	ProductName	Color	ModelName	OrderQuantity	UnitPrice
2	Road Bikes	Bikes	BK-R93R-48	Road-150 Red, 48	Red	Road-150	1	3578.27
3	Road Bikes	Bikes	BK-R93R-62	Road-150 Red, 62	Red	Road-150	1	3578.27
4	Mountain Bikes	Bikes	BK-M82S-38	Mountain-100 Silver, 38	Silver	Mountain-100	1	3399.99
5	Mountain Bikes	Bikes	BK-M82S-38	Mountain-100 Silver, 38	Silver	Mountain-100	1	3399.99
6	Road Bikes	Bikes	BK-R93R-62	Road-150 Red, 62	Red	Road-150	1	3578.27
7	Road Bikes	Bikes	BK-R93R-62	Road-150 Red, 62	Red	Road-150	1	3578.27
8	Road Bikes	Bikes	BK-R93R-62	Road-150 Red, 62	Red	Road-150	1	3578.27
9	Road Bikes	Bikes	BK-R93R-62	Road-150 Red, 62	Red	Road-150	1	3578.27
10	Road Bikes	Bikes	BK-R93R-62	Road-150 Red, 62	Red	Road-150	1	3578.27
11	Road Bikes	Bikes	BK-R93R-56	Road-150 Red, 56	Red	Road-150	1	3578.27
12	Road Bikes	Bikes	BK-R93R-56	Road-150 Red, 56	Red	Road-150	1	3578.27

RYСУNEK 1-1 Przykładowe dane, które wykorzystamy do utworzenia tabeli przestawnej.

Teraz, gdy mamy już w arkuszu wszystkie potrzebne dane, możemy dołączyć do nich tabelę przestawną, klikając przycisk **PivotTable** (Tabela przestawna) na karcie **Wstawianie** wstążki programu Excel. Wywołany kreator zażąda wskazania tabeli używanej jako źródło i zapyta, gdzie umieścić wynikową tabelę przestawną, po czym udostępni standardowy interfejs tabeli przestawnej pokazany na rysunku 1-2.



RYСУNEK 1-2 Standardowy interfejs tabeli przestawnej w programie Excel.

Tutaj, aby wskazać choć jeden przykład, możemy wybrać na przykład pole *Year* (rok) i umieścić je jako kolumny, *ProductCategory* (kategoria produktów) jako wiersze i wyświetlić *SalesAmount* (Wielkość sprzedaży) na przecięciu kolumn i wierszy. Po odpowiednim sformatowaniu liczb uzyskujemy ładny raport (pokazany na rysunku 1-3), ukazujący wyniki dla każdej kategorii produktów w poszczególnych latach.

Sum of SalesAmount		Column Labels			
Row Labels	2005	2006	2007	2008	Grand Total
Bike Racks			16,440.00	22,920.00	39,360.00
Bike Stands			18,921.00	20,670.00	39,591.00
Bottles and Cages			23,280.27	33,517.92	56,798.19
Caps			7,956.15	11,731.95	19,688.10
Cleaners			3,044.85	4,173.75	7,218.60
Fenders			19,408.34	27,211.24	46,619.58
Gloves			14,228.69	20,792.01	35,020.70
Helmets			92,583.54	132,752.06	225,335.60
Hydration Packs			16,771.95	23,535.72	40,307.67
Jerseys			70,370.46	102,580.22	172,950.68
Mountain Bikes	585,973.27	1,562,456.76	3,989,638.48	3,814,691.06	9,952,759.56
Road Bikes	2,600,400.39	4,967,086.77	3,952,029.21	2,920,267.67	14,520,504.04
Shorts			30,445.65	40,874.16	71,319.81
Socks			2,229.52	2,876.80	5,106.32
Tires and Tubes			103,259.76	142,269.56	245,529.32
Touring Bikes			1,417,434.93	2,427,366.12	3,844,801.05
Vests			13,017.50	22,669.50	35,687.00
Grand Total	3,266,373.66	6,530,343.53	9,791,060.30	9,770,899.74	29,358,677.22

RYСУNEK 1-3 Oto przykład raportu utworzonego przy użyciu tabeli przestawnej.

Jasno widać, że zmieniając sposób organizacji danych w wiersze i kolumny możemy łatwo tworzyć rozmaite interesujące raporty, posługując się intuicyjnym i szybkim interfejsem, ułatwiającym poruszanie się wśród informacji.

Rysunek 1-3 pokazuje, jak wygląda standardowa tabela przestawna. Ludzie na całym świecie z powodzeniem korzystają z tego narzędzia od wielu lat, analizując swoje dane na rozmaite sposoby i tworząc raporty odpowiadające ich potrzebom.

Jedną z najważniejszych (i najlepszych) cech narzędzia tabel przestawnych jest prostota użytkowania. Excel sam analizuje źródłową tabelę, wykrywa wartości numeryczne i zapewnia możliwość podsumowania danych częściowych ze wszystkich kolumn. Łatwo zauważyć, że podsumowania są tworzone przy użyciu funkcji *SUMA*¹ (*SUM*), gdyż to właśnie jest najczęściej potrzebne. Można jednak użyć innej funkcji agregującej, posługując się rozmaitymi opcjami tabeli przestawnej.

¹ Ze względu na fakt, że w Polsce zdecydowanie częściej spotykana jest polska wersja programu Excel, wszystkie polecenia i funkcje programu zostały przedstawione w polskiej wersji. Przy pierwszym użyciu jakiejś funkcji, nazwy polecenia, opcji menu lub okna dialogowego w nawiasie została podana odpowiednia nazwa występująca w angielskiej wersji programu. Lista funkcji programu Excel w wersji angielskiej i odpowiadające im polecenia w wersji polskiej znajduje się w Dodatku na końcu książki. (Wszystkie przypisy pochodzą od tłumacza).

Przy całej prostocie stosowania, tabele przestawne mają jednak pewne ograniczenia:

- Tabele przestawne umożliwiają analizę informacji zawartych w jednej tylko tabeli przechowywane w arkuszu Excela. Jeśli mamy wiele różnych arkuszy, zawierających różne informacje, nie istnieje prosta metoda skorelowania informacji znajdujących się w tych arkuszach.
- Nie zawsze łatwe jest uzyskanie danych źródłowych w formacie nadającym się do analizy. W poprzednim przykładzie pokazaliśmy tabelę uzyskaną poprzez kwerendę SQL wykonaną w bazie danych *AdventureWorks*, zbudowaną specjalnie w celu analizy danych. Umiejętności potrzebne do utworzenia takiej kwerendy nie są powszechne i charakterystyczne raczej dla personelu technicznego, gdyż konieczna jest znajomość nie tylko składni języka SQL, ale również struktury odpytywanej bazy danych. Często okazuje się, że konieczne jest poproszenie personelu IT o utworzenie takiej kwerendy, zanim w ogóle będzie można rozpocząć analizę.
- Ponieważ na raz można analizować tylko jedną tabelę, często powstaje konieczność tworzenia kwerend adresowanych dla określonego typu analizy, a jeśli z jakiegoś powodu będziemy chcieli wykonać analizę innych aspektów naszych danych, trzeba będzie zbudować inne kwerendy. Na przykład, jeśli mamy kwerendę zwracającą wielkość sprzedaży na poziomie miesięcy, nie możemy jej użyć do wykonania pogłębionych analiz na poziomie „dnia tygodnia”. Do tego niezbędna będzie nowa kwerenda. Może to wymagać ponownego kontaktu z działem IT, co zawsze pociągnie za sobą jakieś koszty – chociażby stracony czas.

Gdy tabele przestawne już nie wystarczają, co często jest spotykane w przedsiębiorstwach średnich rozmiarów, często podejmowana jest decyzja o rozpoczęciu kompletnego projektu BI, wykorzystującego takie produkty jak SQL Server Analysis Services, które zapewniają analogiczne funkcje przekształcania danych w bardzo złożonych strukturach, znanych jako *kostki OLAP*. Są one stosunkowo trudne do zbudowania, ale są najlepszym rozwiązaniem, jeśli potrzebujemy możliwości swobodnej, nieograniczonej analizy przechowywanych danych. Kostki OLAP zostaną skrótowo przedstawione w dalszej części tej książki w rozdziale 4, „Istota modeli danych”. Na razie wystarczy stwierdzenie, że wprawdzie stanowią one ostateczne i pełne rozwiązanie potrzeb analizy biznesowej, ale są kosztowne i wymagają znacznego wysiłku ze strony działu IT.

Korzystanie z PowerPivot w Microsoft Office 2013

Tabele przestawne oparte na standardowych tabelach Excela to wygodne, poręczne narzędzie. Tym niemniej, aby umożliwić analizowanie bardziej złożonych struktur danych, firma Microsoft wprowadziła funkcjonalność nazywaną „samoobsługowym BI”. Celem tego rozwiązania jest umożliwienie budowania złożonych struktur danych i ich analizy poprzez interfejs analogiczny do tabel przestawnych, ale bez bieżących ograniczeń tego narzędzia. PowerPivot jest podstawowym narzędziem pozwalającym na realizację samoobsługowego BI, wraz z towarzyszącym mu narzędziem Power View, które przedstawimy w dalszej części tego rozdziału.

PowerPivot pozwala użytkownikowi analizować dane bez konieczności proszenia pracowników działu IT o utworzenie niezbędnych, złożonych kwerend. Co więcej, usuwa typowe dla tabel przestawnych ograniczenie do analizy tylko jednej tabeli na raz, gdyż pozwala na odpytywanie wielu tabel jednocześnie, tworząc raporty integrujące informacje pochodzące z różnych źródeł.

W roku 2010 opublikowana została wersja 1.0 PowerPivot for Excel, jako rozszerzenie (add-in) programu Excel 2010. PowerPivot jest w istocie kolumnową bazą danych, która nie korzysta z klasycznych tabel programu Excel. Zamiast tego dane przechowywane są we własnym, wewnętrznym formacie i można się do nich odwoływać poprzez zapytania w języku DAX lub poprzez narzędzia tabel przestawnych. Choć informacje te mogą wydawać się tylko ciekawostkami na temat historii PowerPivot, są one naprawdę ważne: aby PowerPivot mógł działać, dane *nie powinny* być składowane w tabelach Excel, ale muszą być umieszczone wewnątrz bazy danych PowerPivot. Trzeba zapamiętać ten fakt – okaże się przydatny w późniejszym czasie.



UWAGA Baza PowerPivot jest również nazywana „modelem danych programu Excel”. Obydwa terminy odnoszą się do tej samej technologii: model danych programu Excel jest w istocie bazą danych PowerPivot, zaś baza ta jest zapisana w pliku skoroszytu Excela. W tej książce będziemy posługiwać się obiema nazwami w zależności od kontekstu. Jeśli uznamy, że ważne jest odróżnienie PowerPivot od Excela, będziemy mówili o bazie danych PowerPivot. W innych przypadkach będziemy trzymać się bardziej standardowej terminologii i pisać o modelu danych.

Początkowo bazy danych PowerPivot były w jakimś stopniu odseparowane od Microsoft Office, co należy rozumieć tak, że ich funkcje były dostępne tylko dla tych użytkowników, którzy zdecydowali się na pobranie i zainstalowanie rozszerzenia. Jeśli skoroszyt Excel zawierający dane PowerPivot został otwarty na komputerze, na którym rozszerzenie nie zostało zainstalowane, po prostu nie działał, choć dane zawarte w arkuszach były cały czas widoczne.

Posługiwanie się przykładową bazą *AdventureWorks*

Wszystkie przykłady i ćwiczenia zaprezentowane w tej książce wykorzystują przykładową bazę danych *AdventureWorks*. Zdecydowaliśmy się na ten wybór, gdyż baza ta jest szeroko znana, dostępna bezpłatnie i zawiera dostatecznie złożone dane, aby móc wykonywać kompleksowe analizy. Baza ta zawiera dane na temat Adventure Works Cycles, będącej fikcyjną, wielonarodową korporacją produkującą i sprzedającą rowery różnych typów na rynkach Ameryki Północnej, Europy i Azji.

Bazę danych AdventureWorks można pobrać z witryny <http://www.codeplex.com/SqlServerSamples>, w której dostępne są różne wersje bazy dla rozmaitych wydań Microsoft SQL Server. Jeśli Czytelnik nie dysponuje oprogramowaniem SQL Server zainstalowanym na swoim komputerze, może użyć wersji bazy danych w formacie Microsoft Access, dołączonej do materiałów towarzyszących książce. Co więcej, wszystkie przykłady prezentowane w książce są dostępne jako skoroszyty programu Excel. Dzięki temu Czytelnik może wykonać większość przykładów i ćwiczeń nawet wtedy, gdy nie ma dostępu do bazy danych.

Dla bardziej zainteresowanych Czytelników firma Microsoft udostępnia dane przykładowe w postaci skoroszytów Excel, których można użyć do testowania narzędzia PowerPivot. Pliki te można pobrać ze strony <http://tinyurl.com/PowerPivotSamples>. Mimo że nie korzystamy z tych przykładów w tej książce, uważamy, że warto je pobrać i wykorzystać do przeprowadzenia różnorodnych prób i testów.

W wydaniu Office 2013 PowerPivot jest wstępnie instalowany i musi zostać tylko aktywowany. Co ważniejsze, silnik bazy danych PowerPivot został w pełni zintegrowany z kodem programu Excel i działa nawet przed aktywowaniem. Niektóre funkcje są dostępne natychmiast, podczas gdy inne trzeba uaktywnić ręcznie, co pokażemy w dalszej części tego rozdziału.

Aby rozpocząć korzystanie z PowerPivot, zamierzamy wybrać najprostszą drogę: utworzymy tabele PowerPivot (przypomnijmy – są one czymś innym, niż tabele programu Excel) nawet bez aktywowania rozszerzenia. Odbywa się to prosto, gdy tylko uaktywnimy jedną z zaawansowanych funkcji analizy, takich jak:

- Raporty Power View
- Relacje pomiędzy tabelami
- Tabele przestawne odwołujące się do więcej niż jednej tabeli danych

Dodawanie informacji do tabeli programu Excel

Zajmijmy się teraz nieco bardziej złożonymi analizami. Zbiór danych zawartych w naszej tabeli Excela zawiera informacje o kategoriach produktów. Przyjmijmy, że w firmie Adventure Works każda kategoria produktów jest przypisana konkretnemu sprzedawcy i że informacja ta nie została zapisana w bazie danych, zatem nie możemy po prostu zmodyfikować oryginalnej kwerendy, aby dołączyć te informacje. Możemy jednak utworzyć dodatkową tabelę Excela, wypełniając ją tymi informacjami, co prezentuje rysunek 1-4.

Category	SalesManager	Office
Bike Stands	Maurizio Macagno	Redmond
Bike Stands	Maurizio Macagno	Redmond
Bottles and Cages	Maurizio Macagno	Redmond
Caps	Maurizio Macagno	Redmond
Cleaners	Alberto Ferrari	Seattle
Fenders	Alberto Ferrari	Seattle
Gloves	Alberto Ferrari	Seattle
Helmets	Alberto Ferrari	Seattle
Hydration Packs	Alberto Ferrari	Seattle
Jerseys	Mario Russo	Redmond
Mountain Bikes	Mario Russo	Redmond
Road Bikes	Mario Russo	Redmond
Shorts	Mario Russo	Redmond
Socks	Mario Russo	Redmond
Tires and Tubes	Mario Russo	Redmond
Touring Bikes	Louis Bonifaz	Seattle
Vests	Louis Bonifaz	Seattle

RYСУNEK 1-4 Tabela SalesManager okaże się przydatna, gdy zechcemy zademonstrować wydajność menedżera kategorii produktu, a nie samych kategorii.

Aby móc użyć tych informacji w tabeli przestawnej, musimy wprowadzić do oryginalnego modelu danych kolumnę *SalesManager* (menedżer sprzedaży). Do wypełnienia jej, co Czytelnik zapewne już wie, nieoceniona będzie funkcja *WYSZUKAJ.PIONOWO* (*VLOOKUP*). Dodajemy zatem do oryginalnych danych kolumnę zawierającą poniższą formułę²:

```
=WYSZUKAJ.PIONOWO([@ProductCategory];SalesManagers;2)
```

W efekcie uzyskamy nowy zbiór danych zawierających nazwisko menedżera kategorii produktów, jak na rysunku 1-5.

2 Trzeba zwrócić uwagę, że przy korzystaniu z polskiej wersji programu Excel separatorem argumentów funkcji jest średnik (;), podczas gdy w wersji angielskiej jest to przecinek (.). Ponadto symbolem dziesiętnym w angielskiej wersji jest kropka, zaś w wersji polskiej przecinek. Nie stanowi to jednak problemu, gdyż Excel automatycznie dokonuje zamiany separatorów (i nazw funkcji) przy otwieraniu arkusza utworzonego w innej wersji językowej. Przykłady kodu pokazane w książce zostały skorygowane, aby odzwierciedlały składnię używaną w polskiej wersji programu.

B2 =VLOOKUP([@ProductCategory],SalesManagers,2)

	A	B	C	D
1	ProductCategory	SalesManager	ProductSubcategory	ProductCode
2	Road Bikes	Marco Russo	Bikes	BK-R93R-48
3	Road Bikes	Marco Russo	Bikes	BK-R93R-62
4	Mountain Bikes	Marco Russo	Bikes	BK-M82S-38
5	Mountain Bikes	Marco Russo	Bikes	BK-M82S-38
6	Road Bikes	Marco Russo	Bikes	BK-R93R-62
7	Road Bikes	Marco Russo	Bikes	BK-R93R-62
8	Road Bikes	Marco Russo	Bikes	BK-R93R-62
9	Road Bikes	Marco Russo	Bikes	BK-R93R-62
10	Road Bikes	Marco Russo	Bikes	BK-R93R-62
11	Road Bikes	Marco Russo	Bikes	BK-R93R-56
12	Road Bikes	Marco Russo	Bikes	BK-R93R-56
13	Mountain Bikes	Marco Russo	Bikes	BK-M82B-38
14	Road Bikes	Marco Russo	Bikes	BK-R93R-56
15	Mountain Bikes	Marco Russo	Bikes	BK-M82B-44

RYSunek 1-5 Przy użyciu funkcji WYSZUKAJ.PIONOWO mogliśmy dodać informację o menedżerze do oryginalnej tabeli.

Dysponując tym zmodyfikowanym zbiorem danych możemy łatwo utworzyć nową tabelę przestawną, umieszczając pole SalesManager w grupie wierszy. Rezultatem jest pożądaný raport, widoczny na rysunku 1-6.

Sum of SalesAmount		Column Labels			
Row Labels		2005	2006	2007	2008 Grand Total
Accessories					
Cleaners			1,466,088.24	2,000,866.00	3,466,954.24
Fenders			3,044.05	4,331.25	7,375.30
Gloves			19,400.34	24,233.24	43,633.58
Helmets			14,220.69	20,792.01	35,012.70
Hydration Packs			92,583.54	132,652.06	225,235.60
			16,771.25	23,533.72	40,304.97
Components					
Touring Bikes			1,437,434.93	2,427,366.32	3,864,801.25
Vests			13,037.50	22,669.50	35,707.00
Information Systems					
Jerseys			70,370.46	102,580.22	1,72,950.68
Mountain Bikes			3,962,436.76	3,034,693.06	9,997,129.82
Road Bikes			4,967,086.77	3,925,029.21	14,520,304.04
Shorts			30,423.65	40,074.16	70,497.81
Socks			2,229.52	2,076.00	4,305.52
Tires and Tubes			103,239.76	142,269.56	245,509.32
Information Systems					
Bike Racks			16,490.00	22,920.00	39,410.00
Bike Stands			10,921.00	20,670.00	31,591.00
Bottles and Cages			23,200.27	33,537.92	56,738.19
Caps			7,936.35	11,731.25	19,667.60
Grand Total		3,266,373.66	6,530,343.53	9,791,060.30	19,587,777.49

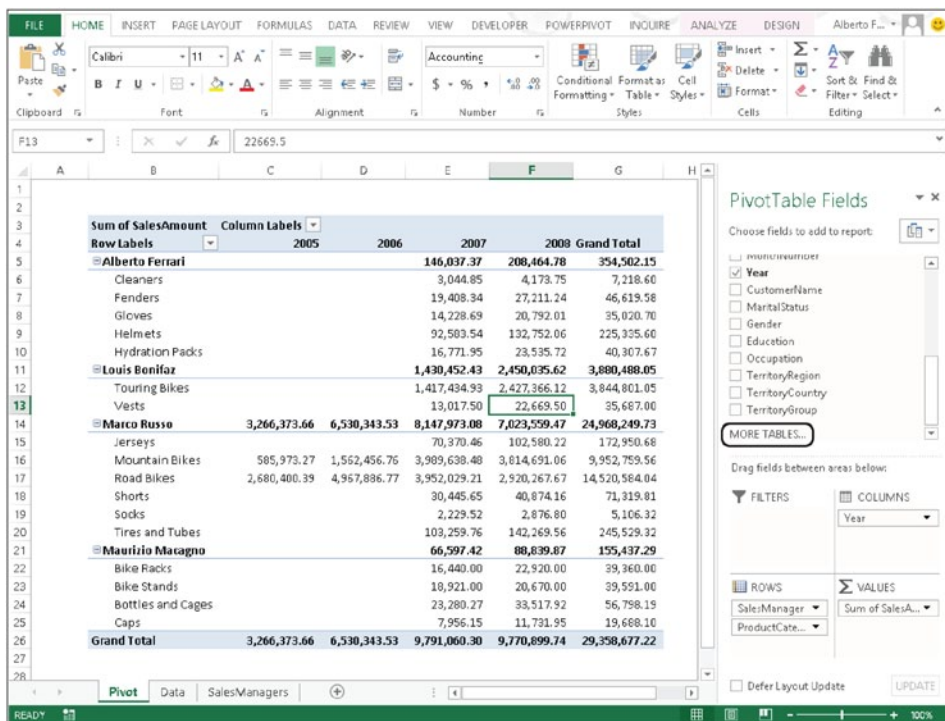
RYSunek 1-6 Dane menedżerów sprzedaży są teraz widoczne w tabeli przestawnej.

Opisana technika sprawdza się dobrze, ale jeśli zechcielibyśmy uzyskać inne pogrupowanie informacji – na przykład na podstawie kolumny Office (biuro) z tabeli

SalesManagers – musielibyśmy powtórzyć całą operację, czyli ponownie posłużyć się funkcją *WYSZUKAJ.PIONOWO* do umieszczenia kolumny Office w oryginalnej tabeli. I nawet jeśli w tym szczególnym przypadku nie jest to wielki wysiłek, znacznie lepszym podejściem będzie przejście na wyższy poziom i poznanie niektórych nowych funkcjonalności programu Excel 2013.

Tworzenie modelu danych zawierającego wiele tabel

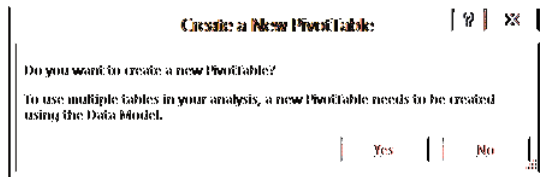
Zamiast dołączania nowej kolumny do zbioru i wykorzystywania funkcji *WYSZUKAJ.PIONOWO* do jej wypełnienia, jak w poprzednim przykładzie, znacznie wygodniej byłoby dodać całą tabelę SalesManagers do tabeli przestawnej, tak by można było użyć wszystkich jej kolumn. Przejdźmy zatem od klasycznej analizy pojedynczej tabeli do bardziej zaawansowanych rozwiązań wykorzystujących wiele tabel. Osiągnięcie tego jest bardzo proste. W dolnej części listy pól tabeli przestawnej znajdziemy opcję **WIĘCEJ TABEL...** (MORE TABLES...), widoczną na rysunku 1-7.



RYСУNEK 1-7 Opcja MORE TABLES... pozwala dołączać wiele tabel do pojedynczego raportu tabeli przestawnej.

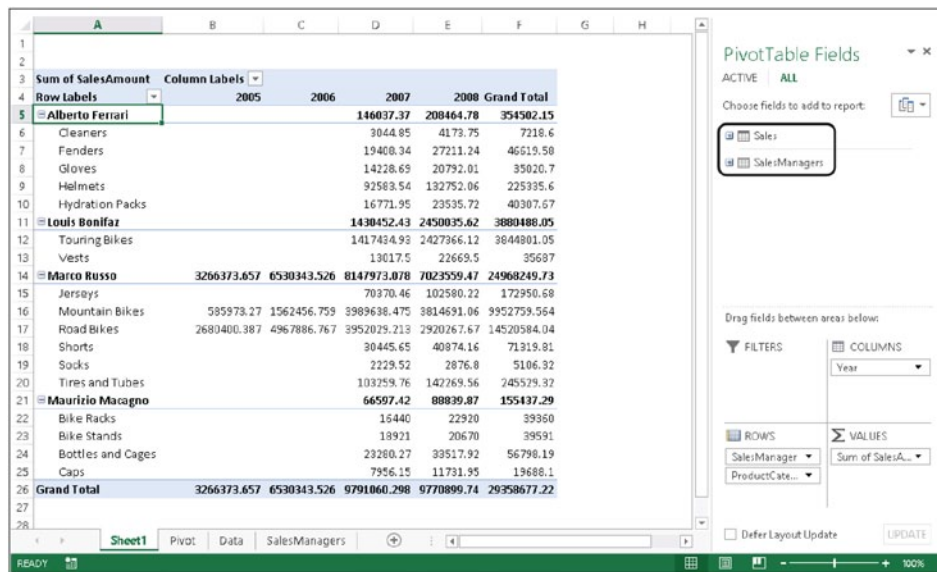
Po kliknięciu opcji **WIĘCEJ TABEL...** ujrzymy okno komunikatu z pytaniem, czy chcemy utworzyć nową tabelę przestawną. Okno to, pokazane na rysunku 1-8, zawiera nieco

interesujących informacji o tym, co się dzieje, w tym nawiązanie do nowego terminu – modelu danych (Data Model).



RYСУNEK 1-8 Pomimo pozornej prostoty, to okno komunikatu zawiera sporo użytecznych informacji.

Gdy klikniemy Tak (Yes), Excel utworzy nową tabelę przestawną o strukturze identycznej z dotychczasową, ale zawierającą więcej tabel danych. Można to zauważyć na rysunku 1-9, gdzie lista dostępnych pól zawiera obecnie dwie tabele.



RYСУNEK 1-9 Nowa tabela przestawna zawiera dwie tabele danych w liście pól.

Teraz możemy usunąć pola SalesManager oraz ProductCategory z listy pól wierszy i po rozwinięciu tabeli SalesManagers dodać do wierszy pole Office. Wynik jednak *nie* jest zgodny z oczekiwaniem. Jak można zauważyć na rysunku 1-10, wszystkie biura oddziałowe (dwa w tym przypadku) prezentują dokładnie te same wartości sprzedaży, co naturalnie nie jest poprawne. Można również zauważyć, że mechanizm tabeli przestawnej sam wykrył błąd, gdyż na liście pól pojawia się komunikat ostrzeżenia „Mogą być potrzebne relacje między tabelami” (Relationships between tables may be needed). Mamy tu również zachęcający przycisk UTWÓRZ... (Create).

Sum of SalesAmount	Column Labels	2005	2006	2007	2008	Grand Total
Row Labels						
Redmond		3266373.657	6530343.526	9791060.298	9770899.74	29358677.22
Seattle		3266373.657	6530343.526	9791060.298	9770899.74	29358677.22
Grand Total		3266373.657	6530343.526	9791060.298	9770899.74	29358677.22

RYSUNEK 1-10 Dodanie pola Office do tabeli przestawnej zwraca nieprawidłowe wyniki i ostrzeżenie o braku relacji.

Jak można się domyślić, utworzenie relacji jest kluczem do tego, aby mechanizm tabeli przestawnej był w stanie pokazać właściwe wartości. Jednak zanim to zrobimy, warto dowiedzieć się nieco więcej o tym, czym są relacje.

Istota relacji

W tym momencie mamy dwie tabele: *Sales* (w przykładowym skoroszycie umieszczona w arkuszu *Data*) oraz *SalesManagers*. Każdy wiersz w tabeli *Sales* dotyczy konkretnego produktu, któremu przypisana jest określona kategoria produktów. Z kolei każdej kategorii odpowiada ustalony menedżer i to powiązanie zawarte jest w tabeli *SalesManagers*. Aby dołączyć nazwisko menedżera do danych sprzedaży, posłużyliśmy się wcześniej funkcją `WYSZUKAJ.PIONOWO`, odnajdując nazwę kategorii w tabeli *SalesManager* i pobierając odpowiadającego jej menedżera.

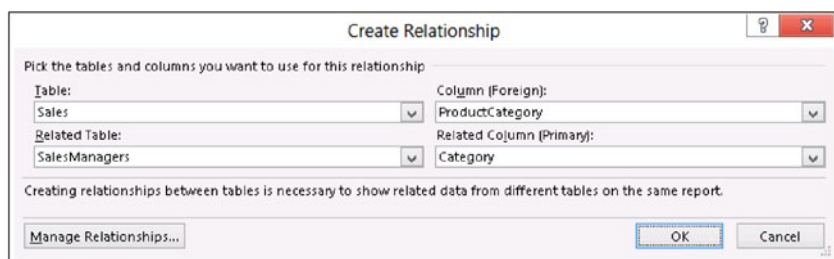
Używając bardziej technicznego języka, możemy zauważyć, że pomiędzy tabelami *Sales* a *SalesManagers* istnieje *relacja* oparta na kolumnie kategorii produktu. Mówiąc precyzyjniej, relacja definiowana jest następująco:

- **Tabela źródłowa** Tabela danych, w której zaczyna się relacja. W tym przykładzie jest to tabela *Sales*, zawierająca kolumnę *ProductCategory*.
- **Kolumna klucza** Kolumna w tabeli źródłowej, zawierająca wartość do wyszukania. W tym przykładzie jest to kolumna *Category*, czyli kategoria produktu, której użyliśmy wcześniej jako parametru funkcji `WYSZUKAJ.PIONOWO`.

- **Powiązana tabela** Tabela zawierająca wartości, których będziemy szukać. W tym przykładzie jest to SalesManagers, zawierająca nazwę kategorii produktu wraz z nazwiskiem menedżera sprzedaży oraz biuro, w którym pracuje ta osoba.
- **Powiązana kolumna** Nazwa kolumny w tabeli powiązanej, zawierająca wartości, które mają zostać dopasowane do kolumny klucza. W naszym przykładzie jest to kolumna Category w tabeli SalesManagers.

O relacji można myśleć jak o pewnego typu automatycznym wyszukiwaniu. W istocie parametry relacji są bardzo podobne do parametrów funkcji *WYSZUKAJ.PIONOWO*. Jediną brakującą informacją jest nazwa odczytywanej kolumny, gdyż po utworzeniu relacji pozwala ona odczytać dowolną kolumnę w powiązanej tabeli bez potrzeby wskazywania jej położenia (co było niezbędne w przypadku funkcji *WYSZUKAJ.PIONOWO*, jako że funkcja ta pozwala odczytać tylko jedną kolumnę z tabeli źródłowej).

Dysponując tymi wskazówkami, klikamy przycisk **UTWÓRZ...** i tworzymy relację, wypinając okno dialogowe, tak jak na rysunku 1-11.



RYСУNEK 1-11 Poprawne parametry tworzenia relacji pomiędzy tabelami w naszym przykładzie.

UWAGA PowerPivot for Excel 2010, poprzednia wersja tego dodatku, dysponował mechanizmem, który automatycznie wykrywał powiązania, co ułatwiało życie w wielu przypadkach. Niestety algorytm detekcji używał mechanizmów heurystycznych do wykrywania powiązań i w niektórych (nie aż tak rzadkich) przypadkach tworzył nieprawidłowe relacje. Z tego względu automatyczna detekcja została usunięta z wersji Excel 2013. Definiowanie relacji jest zadaniem użytkownika. Choć takie podejście może wydawać się krokiem wstecz, jest w rzeczywistości pożądanym rozwiązaniem, gdyż lepiej zachować ostrożność przy tworzeniu relacji, a ludzki mózg działa znacznie lepiej, niż heurystyczne algorytmy.



Kliknięcie **OK** spowoduje, że Excel utworzy relację i zaktualizuje zawartość tabeli przedstawnej, która teraz zawiera właściwe liczby uporządkowane według biur. Rysunek 1-12 pokazuje rezultat z polem SalesManager dołączonym do obszaru wierszy.

Sum of SalesAmount	Column Labels	2005	2006	2007	2008	Grand Total
Redmond		3,266,373.66	6,530,343.53	8,214,570.50	7,112,399.34	25,123,687.02
Marco Russo		3,266,373.66	6,530,343.53	8,147,973.08	7,023,559.47	24,968,249.73
Maurizio Macagnio				66,597.42	88,039.07	155,437.29
Seattle				1,576,489.80	2,658,500.40	4,234,990.20
Alberto Ferrari				146,037.37	208,464.78	354,502.15
Louis Bonifaz				1,430,452.43	2,458,035.62	3,888,488.05
Grand Total		3,266,373.66	6,530,343.53	9,791,060.30	9,770,899.74	29,358,677.22

RYSUNEK 1-12 Tabela przestawna prezentuje poprawne wartości po zdefiniowaniu relacji.

Relacje odgrywają bardzo ważną rolę w mechanizmie PowerPivot i w dalszej części tej książki dowiemy się znacznie więcej na ich temat. Na razie wystarczy, że będziemy wiedzieć, iż relacje są techniką wiązania dwóch tabel ze sobą przy użyciu wspólnej kolumny. Jeśli dwie kolumny (każda w innej tabeli) zawierają te same wartości dla odpowiednich wierszy, wówczas relacja znajduje wynik i pozwala powiązać pasujące wiersze ze sobą.

Ale... Chwileczkę! Powiedzieliśmy, że relacje są ważne dla PowerPivot? Przecież jak dotąd nie użyliśmy PowerPivot – po prostu posłużyliśmy się funkcjami Excela do utworzenia tabeli przestawnej czerpiącej informacje z więcej niż jednej tabeli. Dlaczego więc ta książka ma PowerPivot w tytule? Przyczyna jest prosta: chociaż nie odwołał się jawnie do PowerPivot, Excel w tle utworzył model danych PowerPivot i „wielozródłowa” tabela przestawna w istocie odwołuje się właśnie do tego modelu. Zatem teraz przyjrzymy się modelowi danych.

Istota modelu danych

Wcześniej na rysunku 1-8 pokazaliśmy okno potwierdzenia zapytaniem, czy utworzyć nową tabelę przestawną przy użyciu modelu danych. Nie wyjaśniliśmy jednak, czym jest model danych, ani dlaczego jest on niezbędny, jeśli chcemy w tabeli przestawnej użyć danych z więcej niż pojedynczego źródła (tabeli). Jasne jest jednak to, że nowa tabela przestawna będzie używać modelu danych. Tym samym interesujące będzie lepsze poznanie istoty modelu danych, zanim zagłębimy się w bardziej zaawansowaną tematykę.

Tabele programu Excel są dokładnie tym, co sugeruje ich nazwa: są tabelami. Możemy mieć setki tabel w skoroszycie Excela, ale każda z nich jest odseparowana od pozostałych. To dlatego przy tworzeniu tabeli przestawnej jesteśmy ograniczeni do pojedynczej tabeli danych źródłowych: dodanie więcej niż jednej tabeli nie ma sensu, gdyż nie mają one żadnego punktu wspólnego. Kluczem do przekształcenia zbioru indywidualnych tabel w *model danych* są relacje. Jeśli wiele tabel zostanie połączonych relacjami, możliwe będzie pokazanie jednocześnie danych pochodzących z wielu źródeł w tabeli przestawnej, gdyż filtrowanie jednej tabeli powoduje przefiltrowanie również innych, powiązanych tabel.

W poprzednim przykładzie umieszczenie filtru w kolumnie Office tabeli SalesManager narzuciło również filtr na tabelę Sales. W istocie, wiersze z informacjami o biurze Seattle pokazują tylko wartości dla tych kategorii produktów, które są obsługiwane przez personel z tego biura. Powodem, dla którego tabela Sales jest filtrowana według kolumny Office (z innej tabeli!), jest to, że każda sprzedaż jest przypisana do menedżera sprzedaży, który jest przypisany do określonego biura. Relacja pomiędzy obiema tabelami sprawia, że ten mechanizm działa. Tak więc prawdziwe są następujące spostrzeżenia:

- Zbiór tabel nie jest niczym więcej, niż zbiorem oddzielnych tabel.
- Zbiór tabel z relacjami wiążącymi je nawzajem stanowi model danych.

W wydaniu Excel 2013 wprowadzono koncepcję modelu danych jako jednego z narzędzi dostępnych dla użytkowników w celu analizowania danych. Każda tabela programu Excel może należeć do modelu danych: jest dołączana automatycznie, gdy zostanie zdefiniowana relacja jej dotycząca, bez względu na to, czy jest tabelą źródłową, czy docelową tej relacji.

Wszystko to wygląda świetnie, ale co z tym wszystkim ma wspólnego PowerPivot. Model danych programu Excel jest w rzeczywistości modelem danych PowerPivot. Kiedykolwiek dołączamy tabelę do modelu danych, w istocie dodajemy tabelę do bazy danych PowerPivot, która funkcjonuje wewnątrz skoroszytu programu Excel.

Model danych PowerPivot i tabela programu Excel są dwoma oddzielnymi bytami. Gdy dodajemy tabelę do modelu danych, nie wykonujemy przekształcenia tabeli Excela na tabelę PowerPivot. W rzeczywistości dane zawarte w tabeli Excela są kopiowane do tabeli PowerPivot. Obie tabele są nadal powiązane, zatem jeśli zmodyfikujemy informacje w oryginalnej tabeli Excela i odświeżymy tabelę przestawną, zmiany zostaną zaimportowane do modelu danych PowerPivot. Jednak z punktu widzenia przechowywania informacji dane są naprawdę zduplikowane: mamy oryginalną tabelę programu Excel oraz kopię w PowerPivot.

Tworzenie modelu danych jest bardzo proste. Następuje to automatycznie, gdy tylko Excel wykryje, że potrzebuje utworzyć model danych, aby zaspokoić potrzeby użytkownika. W takim przypadku Excel przetwarza tabele na model danych PowerPivot, gdy tylko będzie to niezbędne – na przykład by utworzyć tabelę przestawną odwołującą

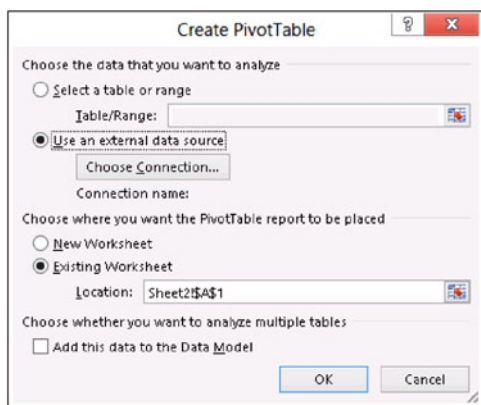
się do więcej niż jednej tabeli danych. Aby zrealizować to zadanie, Excel tworzy model danych PowerPivot, w praktyce eliminując potrzebę pełnego rozumienia tego, co dzieje się pod powierzchnią.

Tym niemniej ważne jest zrozumienie, że posługując się tymi automatycznymi funkcjami wykorzystujemy jedynie nieznaczną część rzeczywistej mocy PowerPivot. Aby móc użyć całej funkcjonalności PowerPivot, musimy nauczyć się pracować bezpośrednio z modelem danych, bez polegania jedynie na automatycznym wykorzystaniu silnika bazy danych PowerPivot, co robi sam Excel.

Odpytywanie modelu danych

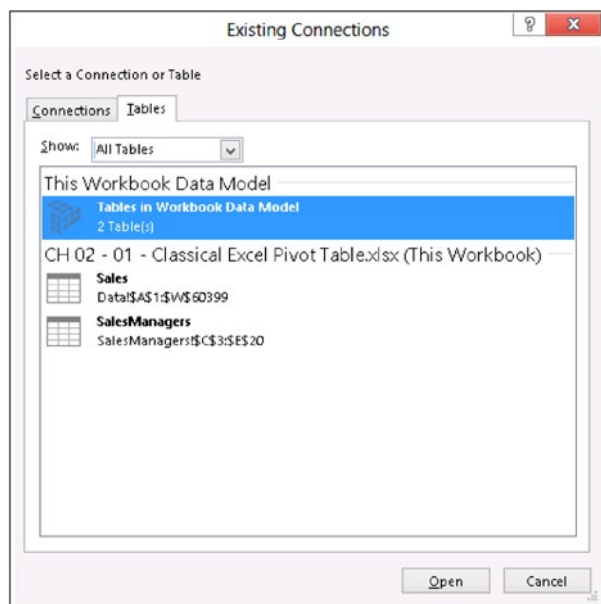
W poprzednim podrozdziale pokazaliśmy, jak, poprzez tworzenie relacji pomiędzy tabelami, możemy zbudować model danych PowerPivot wewnątrz skoroszytu programu Excel. Gdy model danych został już utworzony, można się do niego odwoływać poprzez wiele tabel przestawnych bez konieczności ponownego dodawania tabel do modelu. W tym podrozdziale pokażemy, jak wykonać tę operację, która, choć nie tak łatwo ją znaleźć, jest bardzo wygodna.

Gdy stworzymy nową tabelę przestawną, Excel wyświetla okno dialogowe **Tworzenie tabeli przestawnej** (Create PivotTable), pokazane na rysunku 1-13.



RYСУNEK 1-13 Tworzenie tabeli przestawnej wymaga podania parametrów nowej tabeli.

W tym oknie dialogowym, zamiast wybierać zakres komórek, do czego (zapewne) przyzwyczailiśmy się w przeszłości, należy wybrać opcję **Użyj zewnętrznego źródła danych** (Use an external data source), po czym kliknąć **Wybierz połączenie...** (Choose Connection). Nie jest to zapewne zbyt intuicyjne, skoro nasze dane nadal znajdują się w bieżącym skoroszycie! Tym niemniej, Excel wyświetla listę dostępnych „zewnętrznych” połączeń, których można użyć, w zakładce **Tabele** – znajdziemy tam zarówno poszczególne tabele, jak i model danych, co widać na rysunku 1-14.



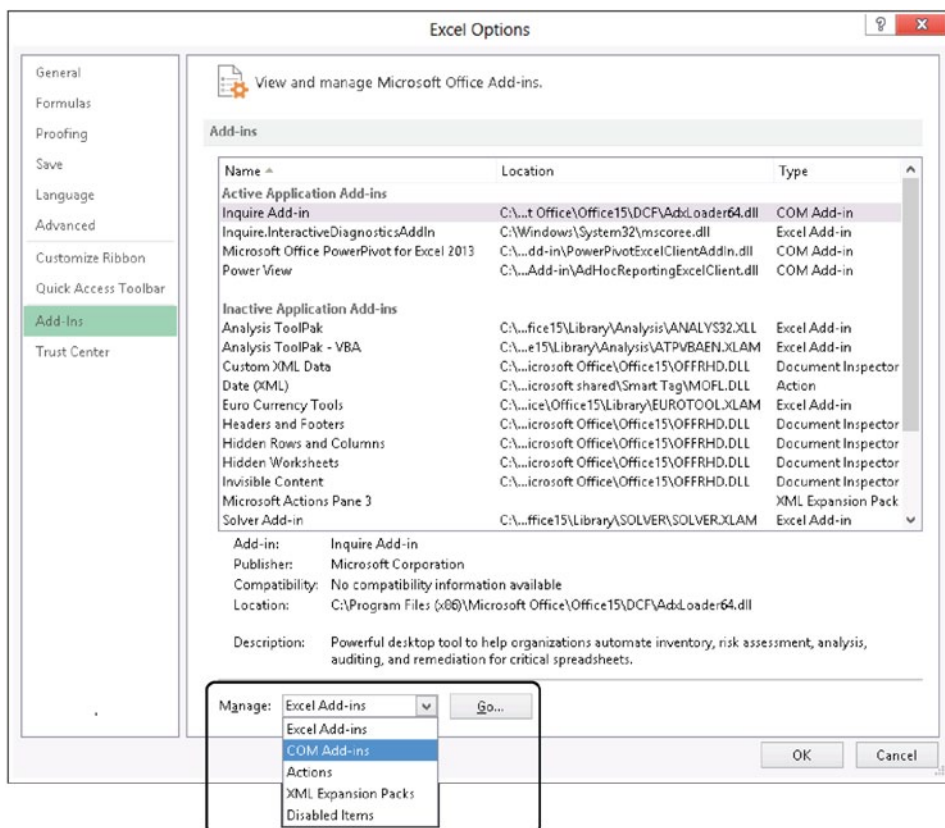
RYСУNEK 1-14 Lista dostępnych źródeł danych zawiera model danych skoroszytu (Workbook), będący zarazem modelem danych PowerPivot.

Wybranie opcji Workbook Data Model i potwierdzenie wszystkich kolejnych kroków w procesie tworzenia tabeli przestawnej powoduje powstanie nowej tabeli, połączonej z tym samym modelem danych utworzonym wcześniej, opartym na oryginalnych tabelach programu Excel.

Dodatek PowerPivot

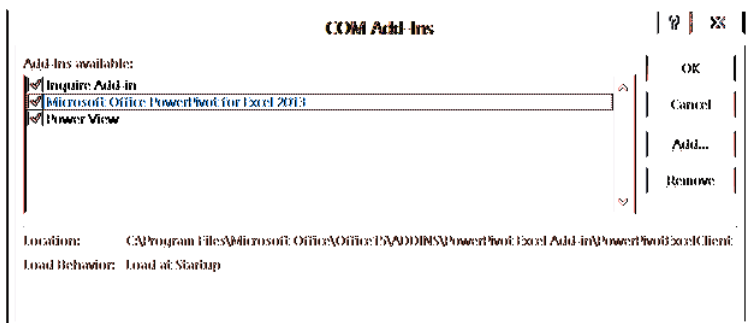
W poprzednich podrozdziałach pokazaliśmy, że nowe funkcje w programie Excel 2013 wymagają utworzenia modelu danych PowerPivot i że model ten może zostać utworzony automatycznie, bez włączania dodatku PowerPivot, który jest wprawdzie instalowany wstępnie, ale początkowo jest wyłączony. Po utworzeniu modelu danych możemy odpytywać go poprzez tabele przestawne (albo, co pokażemy dalej w tym rozdziale, poprzez Power View). Z drugiej strony jednak, jeśli chcemy obejrzeć model danych, Excel nie oferuje sposobu analizowania ani nawet prostego wyświetlenia jego zawartości. Aby móc zobaczyć dane zawarte w modelu, konieczne jest włączenie dodatku PowerPivot i tym zajmiemy się w tym podrozdziale.

W celu włączenia dodatku PowerPivot konieczne jest wejście do opcji programu Excel, wybranie kategorii **Dodatki** (Add-Ins) i wybranie podkategorii **Dodatki COM** (COM Add-Ins) z listy **Zarządzaj** (Manage), jak na rysunku 1-15.



RYSUNEK 1-15 Dodatek PowerPivot musi zostać włączony, aby móc korzystać z jego wszystkich funkcji.

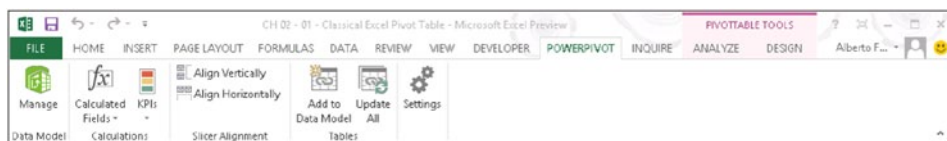
Po wybraniu podkategorii **Dodatki COM** klikamy **Przejdź (Go)**, aby otworzyć listę dostępnych dodatków tego rodzaju, pokazaną na rysunku 1-16.



RYSUNEK 1-16 Lista dostępnych dodatków typu COM umożliwia włączenie lub wyłączenie dodatku PowerPivot.

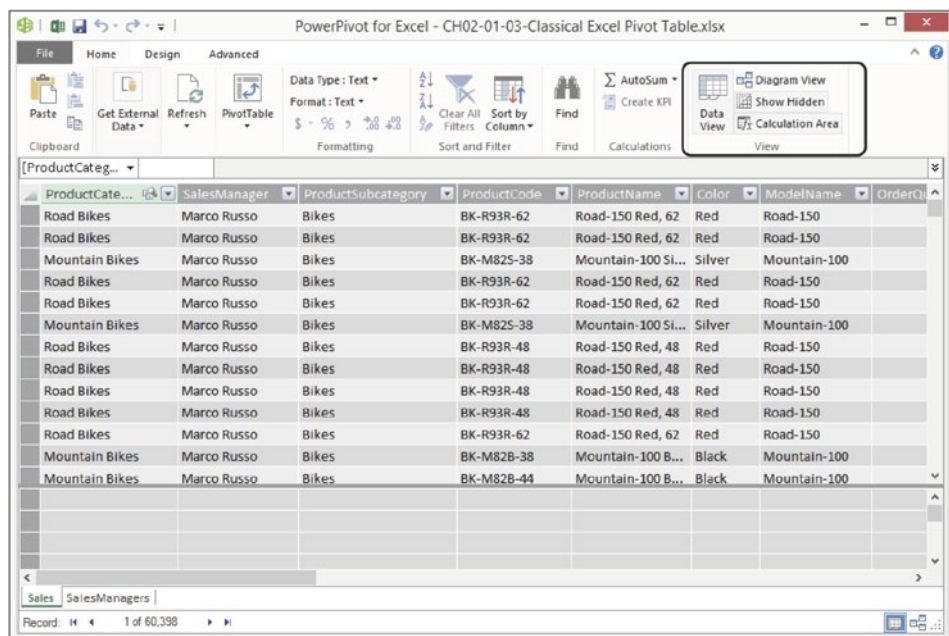
W celu włączenia dodatku trzeba po prostu zaznaczyć pole wyboru obok nazwy Microsoft Office PowerPivot for Excel 2013 i kliknąć OK. Jednak skoro już tu jesteśmy, dobrym pomysłem będzie również włączenie (zaznaczenie pola wyboru) dodatku Power View, który będzie nam potrzebny już niedługo. Power View to inne wspaniałe rozszerzenie możliwości analitycznych programu Excel, które również wykorzystuje model danych PowerPivot.

Po uaktywnieniu dodatku PowerPivot na Wstążce programu Excel zobaczymy nową kartę o nazwie **PowerPivot**, pokazaną na rysunku 1-17.



RYСУNEK 1-17 Karta PowerPivot we Wstążce pojawia się po uaktywnieniu dodatku.

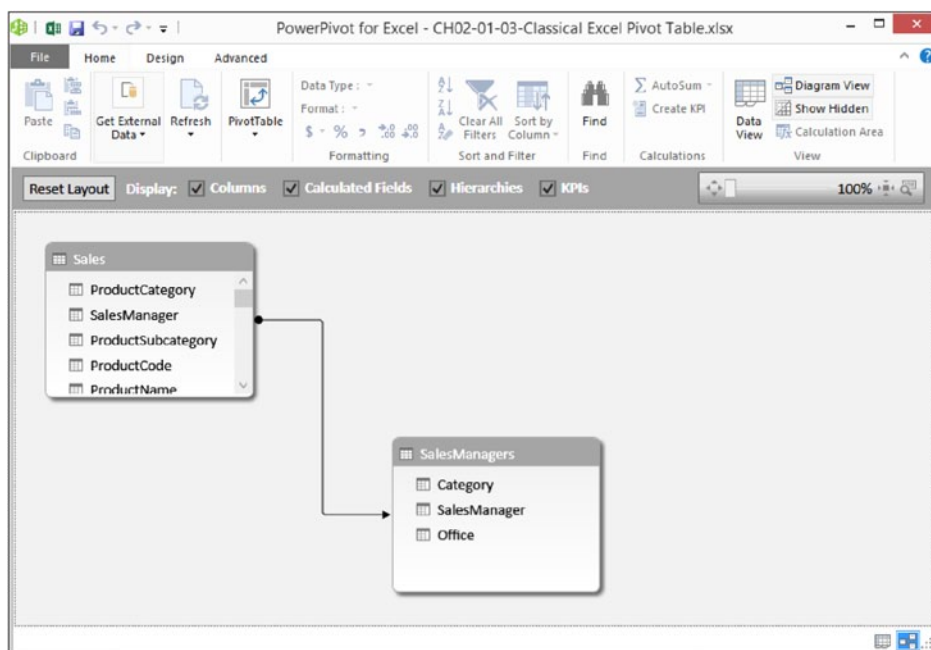
Karta ta udostępnia wiele bardzo interesujących funkcji, które będziemy objaśniać w kolejnych rozdziałach tej książki. Na razie jesteśmy zainteresowani samym otwarciem okna PowerPivot i krótką wycieczką do modelu danych, który właśnie utworzyliśmy. W tym celu należy kliknąć przycisk **Zarządzaj** (Manage) we wstążce, co spowoduje otwarcia okna PowerPivot przedstawionego na rysunku 1-18.



RYСУNEK 1-18 Okno PowerPivot jest głównym miejscem, w którym można skorzystać z zaawansowanych funkcji dodatku.

Okno PowerPivot domyślnie otwiera się w widoku danych, ukazując zawartość tabel PowerPivot dostępnych w modelu. Można tu po prostu przeglądać tabele wiersz po wierszu, a w dolnej części okna ujrzymy zakładki odpowiadające tabelom załadowanym do modelu. Chwilowo nie jesteśmy zainteresowani poznawaniem wszystkich funkcji tego okna. Chcemy jedynie użyć go do szybkiego przyjrzenia się modelowi danych. Aby uzyskać bardziej czytelny obraz, należy kliknąć przycisk **Widok diagramu** (Diagram View) we wstążce okna PowerPivot.

Widok diagramu jest bardzo wygodną metodą wizualizacji modelu danych, gdyż zamiast skupiania się na zawartości tabel, pokazuje strukturę relacji, przez co łatwiej można zrozumieć wzajemne zależności (rysunek 1-19).



RYСУNEK 1-19 Widok diagramu pozwala skupić się na strukturze zależności pomiędzy tabelami w modelu danych.

Widok diagramu stanowi kanoniczną reprezentację typu „ramki i strzałki” modelu relacyjnego, który jest przechowywany w modelu danych. Każda tabela jest reprezentowana przez ramkę, zaś relacje pomiędzy nimi są przedstawione w formie strzałek biegnących od jednej ramki do drugiej. Kliknięcie relacji (czyli reprezentującej ją strzałki) powoduje podświetlenie kolumn, które stanowią części relacji.

Sposoby korzystania z wielu funkcji tego okna poznamy w dalszych częściach książki; na razie możemy je po prostu zamknąć. Poczynając od rozdziału 3, „Wprowadzenie do DAX”, zaczniemy wykorzystywać widok diagramu do modyfikowania modelu

danych. Chwilowo jednak chcemy raczej przedstawić główne funkcje programu Excel 2013 powiązane z PowerPivot, niż omawiać je szczegółowo.

Korzystanie z narzędzi OLAP i konwertowanie na formuły

Jedną z nowych funkcji programu Excel, dostępnych w modelu danych, ale nie w jednотabelowej tabeli przestawnej, jest sekcja **Narzędzia OLAP** (OLAP Tools). Ten zestaw funkcji był oryginalnie dostępny tylko dla tabel przestawnych zbudowanych na podstawie baz danych OLAP (i stąd taka właśnie nazwa), ale ze względu na naturę modelu danych, który w istocie jest bazą danych PowerPivot, funkcja ta jest obecnie dostępna również przy pracy z modelem danych.

Tabele przestawne są naprawdę skutecznymi narzędziami do eksploracji danych. Tym niemniej często stanowią tylko pierwszy krok podczas tworzenia złożonych raportów, które pobierają dane z tabel przestawnych, wykonują różnorodne obliczenia i formatowanie, po czym udostępniają finalne rezultaty w formie zwężonych raportów, niekiedy nazywanych *tablicami kontrolnymi* (*dashboards*). Mówiąc w uproszczeniu, tablica nie jest niczym więcej niż raportem zawierającym wiele elementów informacji (z których każdy pochodzi z kwerendy wykonanej w bazie danych), zapewniającym przejrzystą reprezentację stanu firmy.

Żałóży, że chcemy utworzyć raport zawierający całkowitą sprzedaż, procentowy wzrost sprzedaży oraz procentowy podział pomiędzy kanał Internetowy i resellerski dla ostatnich trzech lat. Raport ten powinien prezentować informacje podzielone według regionów, aby można było ustalić, które obszary wymagają większej uwagi.

Rysunek 1-20 pokazuje taki raport w postaci finalnej; można go znaleźć w skoroszycie *CH01-02-Dashboard.xlsx*.

	Total Sales			Growth		2006		2007		2008	
	2006	2007	2008	2007	2008	Internet	Resell	Internet	Resell	Internet	Resell
Europe	1,478,106	4,757,754	13,553,399	222%	185%	100%		40%	60%	41%	59%
France	414,245	2,061,420	4,772,398	398%	132%	100%		31%	69%	33%	67%
Germany	513,353	593,247	3,768,095	16%	535%	100%		100%		47%	53%
United Kingdom	-	2,103,087	5,012,905		138%			33%	67%	43%	57%
North America	19,313,719	26,826,465	33,182,889	39%	24%	16%	84%	6%	94%	20%	80%
Canada	3,652,908	5,920,180	6,771,829	62%	14%	16%	84%	5%	95%	16%	84%
United States	15,660,811	20,906,285	26,411,060	33%	26%	16%	84%	7%	93%	21%	79%
Central	2,024,176	2,812,658	3,072,175	39%	9%		100%	0%	100%	0%	100%
Northeast	1,182,920	3,354,277	2,402,177	184%	-28%		100%	0%	100%	0%	100%
Northwest	4,181,083	4,006,507	7,887,187	-4%	97%	22%	78%	15%	85%	27%	73%
Southeast	2,857,048	2,483,826	2,538,667	-13%	2%		100%	0%	100%	0%	100%
Southwest	5,415,584	8,249,017	10,510,854	52%	27%	28%	72%	10%	90%	32%	68%
Pacific	2,568,701	2,099,585	5,977,815	-18%	185%	100%		100%		73%	27%
Australia	2,568,701	2,099,585	5,977,815	-18%	185%	100%		100%		73%	27%

RYСУNEK 1-20 Prosty raport typu tablica utworzony z tabeli przestawnej.

Naturalnie nie jesteśmy zainteresowani tymi konkretnie wynikami; to, na czym będziemy się skupiać, to technika tworzenia takiego raportu. Tak więc, mając jasną

koncepcję, jakie efekty chcemy uzyskać, przyjrzyjmy się problemom, które będzie trzeba rozwiązać w tym celu:

- Pierwszy problem dotyczy podziału geograficznego. W gotowym raporcie mamy dane rozbite na grupy, kraje i regiony, ale dla tych krajów, które zawierają tylko jeden region, poziom regionu został usunięty jako bezużyteczny. Jest to coś, czego tabela przestawna nie jest w stanie obsłużyć samodzielnie, choć jest to bardzo częsty wymóg stawiany raportom.
- Nagłówki zawierają całkowitą sprzedaż i wzrost podzieloną według lat (i wybrane lata dla tych dwóch kolumn się różnią, gdyż wzrost nie jest pokazywany dla roku 2006 – pierwszego w tych danych). Co więcej, w nagłówkach kolumn chcemy pokazać wyniki roczne podzielone według kanału sprzedaży (Internet i resellerzy). Tego typu mieszany sposób rozdzielania danych nie jest czymś, co można zrealizować w tabeli przestawnej.
- Dane wewnątrz komórek stanowią połączenie sprzedaży internetowej i resellerskiej w niektórych komórkach (tych zawierających całkowitą sprzedaż); inne komórki pokazują udział sprzedaży internetowej lub resellerskiej w całkowitym wyniku.
- Co więcej, za pośrednictwem opcji formatowania warunkowego dostępnych w Excelu raport pokazuje komórki w różnych kolorach, kierujących uwagę czytelnika do najbardziej interesujących danych. Łatwo zauważyć, że formatowanie to wykorzystuje odmienne formuły dla różnych komórek.
- Nie trzeba mówić, że nie da się utworzyć raportu tego rodzaju przy użyciu dowolnego typu tabeli przestawnej. Tym niemniej, na pewno możemy uzyskać wartości niezbędne do wyliczenia poszczególnych komórek raportu, używając jednej lub kilku tabel przestawnych. Wszystko, co trzeba zrobić, to utworzenie (zapewne kilku) tabel przestawnych wykonujących potrzebne obliczenia i następnie posłużenie się standardowymi formułami Excela do przeniesienia oryginalnych części danych do naszej tablicy.

Budowanie tej tablicy rozpoczniemy od utworzenia tabeli przestawnej pokazanej na rysunku 1-21, używając prostego układu pokazującego wielkości sprzedaży odczytywane z tabeli FactResellerSales.

Ta tabela przestawna, wraz z bardzo podobną prezentującą sprzedaż internetową, zawiera podstawowe informacje potrzebne do utworzenia raportu. Teraz, jako że jest to skoroszyt PowerPivot, ktoś mógłby oczekiwać, że będziemy tworzyć bardzo złożoną formułę w języku DAX, która magicznie zbuduje naszą tablicę, ale byłby w błędzie. Tabele przestawne są użyteczne przy interaktywnej analizie danych, ale nie ma potrzeby zmuszać ich do tego, aby robiły więcej, niż to, do czego zostały wymyślane. Gdy mamy już dostępne potrzebne dane, Excel przeniesie je do wnętrza tablicy i wykona niezbędne obliczenia.

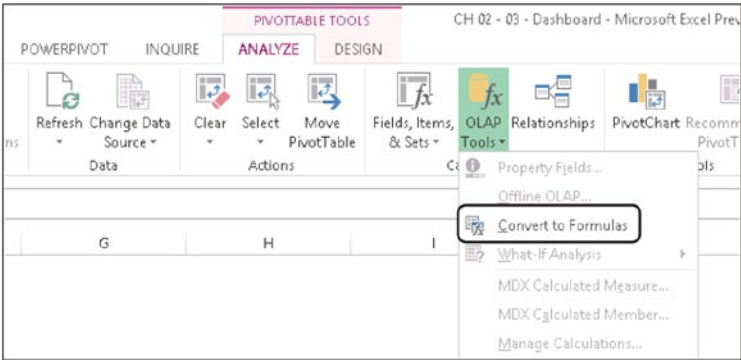
Sum of SalesAmount		Column Labels	
Row Labels	2006	2007	2008
Europe		2,834,512.33	8,036,022.46
France		1,428,020.38	3,179,517.56
Germany			1,983,988.04
United Kingdom		1,406,491.96	2,872,516.87
North America	16,288,441.77	25,087,158.18	26,610,126.86
Canada	3,079,806.81	5,615,169.14	5,682,949.64
United States	13,208,634.95	19,471,989.04	20,927,177.22
Central	2,024,175.80	2,810,586.67	3,071,245.71
Northeast	1,182,920.47	3,352,228.07	2,397,693.48
Northwest	3,255,251.43	3,386,959.00	5,792,064.77
Southeast	2,857,047.77	2,480,188.69	2,530,179.76
Southwest	3,889,239.48	7,442,025.82	7,135,193.50
Pacific			1,594,335.38
Australia			1,594,335.38
Grand Total	16,288,441.77	27,921,670.52	36,240,484.70

RYSunek 1-21 Ta tabela przestawna jest punktem wyjścia dla naszej tablicy.

Można rozpocząć tworzenie naszego raportu-tablicy opartego na dopiero co utworzonej tabeli przestawnej, tworząc formuły, które przeniosą informacje z tabeli przestawnej do tablicy. Nie zapominajmy jednak, że tabele przestawne są dynamiczne z samej swojej natury. Mogą zmienić swoje rozmiary, gdy nowe wartości pojawią się wewnątrz tabel źródłowych. Tym samym, jeśli odwołamy się bezpośrednio do komórek wewnątrz tabeli przestawnej, powstaje ryzyko, że konieczne będzie modyfikowanie tych formuł, gdy dane źródłowe zostaną zaktualizowane. Znacznie lepiej byłoby znaleźć sposób „zablokowania” tabeli przestawnej, aby nie mogła dynamicznie zmieniać rozmiarów.

Szczęśliwie Excel udostępnia opcję, która pomaga przekształcić dynamiczną tabelę przestawną w zbiór statycznych formuł prezentujących te same dane, ale bez dynamicznej natury. Utracimy w ten sposób możliwość wygodnego nawigowania po danych, ale za to uzyskamy unieruchomienie komórek.

Na karcie ANALIZA wstążki Excela znajdziemy przycisk o nazwie **Narzędzia OLAP**, zawierający kilka różnych funkcji. Jedną z nich to **Konwertuj na formuły** (Convert To Formulas), co widać na rysunku 1-22.



RYСУNEK 1-22 Opcja Convert To Formulas dostępna dla tabel przestawnych opartych na modelu danych.

Jeśli wybierzemy tę opcję, tabela przestawna zostanie usunięta i zastąpiona zwykłym arkuszem Excela, zawierającym odpowiednią formułę w każdej komórce (rysunek 1-23).

Sum of SalesAmount		Columns Labels		
Row Labels	2005	2006	2007	2008
Europe			2,834,512.33	8,036,022.46
France			1,428,020.38	3,179,517.56
France			1,428,020.38	3,179,517.56
Germany				1,983,988.04
Germany			1,406,491.96	1,983,988.04
United Kingdom			1,406,491.96	2,872,516.87
United Kingdom			1,406,491.96	2,872,516.87
North America		16,288,441.77	25,087,158.18	26,610,126.86
Canada		3,079,806.81	5,615,169.14	5,682,949.64
Canada		3,079,806.81	5,615,169.14	5,682,949.64
United States		13,208,634.95	19,471,989.04	20,927,177.22
Central		2,024,175.80	2,810,586.67	3,071,245.71
Northeast		1,182,920.47	3,352,228.07	2,397,693.48
Northwest		3,255,251.43	3,386,959.80	5,792,864.77
Southeast		2,857,047.77	2,480,188.69	2,530,179.76
Southwest		3,889,239.48	7,442,025.82	7,135,193.50
Pacific				1,594,335.38
Australia				1,594,335.38
Australia				1,594,335.38
Grand Total		16,288,441.77	27,921,670.52	36,240,484.70

RYСУNEK 1-23 Arkusz, który zawierał tabelę przestawną, po użyciu polecenia Konwertuj na formuły.

Warto teraz przyjrzeć się formułom zawartym w poszczególnych komórkach. Można zauważyć, że występują tu dwa różne typy:

- **ELEMENT.MODUŁU (CUBEMEMBER)** Ten typ formuły jest używany w nagłówkach – zarówno kolumn, jak i wierszy. Zwraca ona obiekt, nazywany *elementem* (*member*), którym jest zasadniczo wartość kolumny w tabeli danych. Na przykład formuła w nagłówku „2008” zawiera:

```
=ELEMENT.MODUŁU("ThisWorkbookDataModel";"[DimTime].[FiscalYear].&[2008]")
```

Można to odczytać następująco: „zwróć wartość kolumny FiscalYear w tabeli DimTime, gdzie wartość wynosi 2008”.

Formuła na pozór jest niezrozumiała, gdyż zwraca wartość „2008” i dostarczyliśmy do niej „2008” jako parametr, co nie wydaje się być sensowne. Dzieje się tak ze względu na wewnętrzne mechanizmy PowerPivot, który „rozumie” w kategoriach elementów i wartości kostki OLAP. Co więcej, warto zauważyć w tym momencie, że składnia używana w wyrażeniach ELEMENT.MODUŁU i WARTOŚĆ.MODUŁU nie jest składnią DAX – jest to MDX. Excel nadal używa języka MDX przy interakcji z PowerPivot.

- **WARTOŚĆ.MODUŁU (CUBEVALUE)** Ten typ formuły jest używany w komórkach wewnątrz tabeli. Każda komórka wywołuje wartość, podając zbiór elementów (wymiarów), które łącznie tworzą współrzędne żądanej wartości. Na przykład w przypadku sprzedaży w Ameryce Północnej dla roku 2008 owe koordynaty to:

❑ `B3 = [Measures].[Sum of SalesAmount]`

❑ `F4 = [DimTime].[FiscalYear].&[2008]`

❑ `B12 = [DimSalesTerritory].[SalesTerritoryGroup].&[North America]`

Rezultatem jest wielkość miary SalesAmount w roku 2008 dla Ameryki Północnej (North America).

Czytelnicy przyzwyczajeni do baz OLAP rozpoznają w tych formułach standardowy zbiór koordynat używany do nawigowania po kostkach OLAP. W istocie, na samym dnie PowerPivot przechowuje swoje dane w kostkach, które są automatycznie przetwarzane przez niego samego, czemu przyjrzymy się bliżej w dalszej części tego rozdziału.

Na razie nie będziemy ciągnąć tej dygresji zbyt daleko. Jedynym interesującym aspektem uzyskanej sytuacji jest to, że arkusz jest obecnie złożony z formuł (tabela przestawna zniknęła) i że te formuły można przenieść w dowolne miejsce. Co więcej, nawet jeśli źródłowe dane się zmieniają, Excel nie zmieni położenia żadnej komórki.

Możemy teraz utworzyć nowy arkusz i zacząć pisać swój raport-tabcicę, odwołując się do wartości wewnątrz tego nowego zbioru komórek. Jedyną rzeczą, którą trzeba mieć na uwadze, jest to, że jeśli kostka nie zwróci żadnych danych, funkcja ELEMENT.MODUŁU zwraca pusty ciąg znaków. Tym samym, ilekroć zachodzi potrzeba odwołania się do jej wartości, konieczne jest zawarcie jej wewnątrz polecenia JEŻELI, tak jak

w komórce Total Sales dla Europy w roku 2006 (rysunek 1-20), która zawiera poniższą formułę:

```
=JEŻELI(Internet!D5="";0;Internet!D5) + JEŻELI(Resellers!D5="";0;Resellers!D5)
```

Tak więc jedynym zastrzeżeniem, które trzeba uwzględnić przy korzystaniu z wartości pochodzących z kostki OLAP jest to, że puste wartości są pustymi ciągami znaków, a nie liczbami. Pozostałą część tablicy można łatwo utworzyć, używając standardowych formuł Excela i nieco formatowania, które to techniki są od dawna znane i dobrze opisane.



UWAGA Wyniki pochodzące z kostek OLAP są zawsze ciągami znaków (typu tekstowego), tak więc funkcja *CZY.PUSTA* (*ISBLANK*) nam nie pomoże.

Naprawdę interesujące w całym tym podrozdziale nie jest budowanie tablicy jako takiej, ale fakt, że poprzez konwersję tabeli przestawnej na formuły możemy użyć danych z tabeli przestawnej jako cegiełek w bardziej złożonym skoroszybie, używającym oryginalnych danych do zwracania wskaźników, porównań i innych informacji, które następnie można dalej przetwarzać używając całej mocy Excela.

Istota PowerPivot dla Excel 2013

Teraz, gdy już po raz pierwszy użyliśmy modelu danych PowerPivot, przed zagłębieniem się w jego wszystkie funkcje i właściwości przydatne będzie poznanie podstawowych informacji, czym jest silnik bazodanowy PowerPivot. Ten podrozdział zapewni podstawy dla zrozumienia tego zagadnienia. Nie jest to bezpośrednio konieczne do korzystania z PowerPivot, przynajmniej na podstawowym poziomie. Jednak wierzymy, że jest to interesujący temat. Można to postrzegać jako niewielką dygresję, zapewniającą przynajmniej ogólną orientację w dziedzinie kolumnowych baz danych.

PowerPivot dla Excel 2013 jest w rzeczywistości instancją SQL Server Analysis Services 2012 Tabular, uruchomionym jako proces wewnątrz programu Excel. Mechanizm PowerPivot nosi nazwę *mechanizmu analitycznego xVelocity* i jest oszczędzającą przestrzeń kolumnową bazą danych działającą całkowicie w pamięci.

Większość baz danych, w tym SQL Server, są bazami zorientowanymi wierszowo, dzięki czemu są łatwe do zrozumienia, gdyż działają w bardzo naturalny sposób: każda tabela bazy danych przechowuje dane wiersz po wierszu, dokładnie tak, jak są one (dane) wyświetlane na ekranie.

Dla przykładu rozważmy tabelę danych:

ID_Author	FirstName	LastName	Blog	Posts
1	Alberto	Ferrari	http://sqlblog.com/blogs/alberto_ferrari	27
2	Maurizio	Macagno	http://adventureworks.com/blogs/mmacagno	43
3	Marco	Russo	http://sqlblog.com/blogs/marco_russo	38

Baza danych zorientowana wierszowo fizycznie przechowuje informacje wiersz po wierszu. Na przykład druga pozycja listy zawiera wszystkie kolumny wiersza opisującego dane Maurizio w taki sposób, jak w poniższej tabeli:

ID wiersza	Dane wiersza
1	1, Alberto, Ferrari, http://sqlblog.com/blogs/alberto_ferrari , 27
2	2, Maurizio, Macagno, http://adventureworks.com/blogs/mmacagno , 43
3	3, Marco, Russo, http://sqlblog.com/blogs/marco_russo , 38

Fizyczna implementacja zależna jest od konkretnego rozwiązania bazodanowego. Na przykład SQL Server dzieli przestrzeń magazynową na strony, a każda z nich zawiera jeden lub więcej wierszy. Inne bazy danych mogą używać odmiennych technik, ale ważne jest to, że dane są przechowywane wierszami.

Mówiąc ogólnie, zorientowana wierszowo baza danych wymaga pełnego skanowania wszystkich wierszy tabeli, jeśli chcemy odpytać wszystkie wartości pojedynczej kolumny (na przykład w celu wyliczenia jakiejś agregacji). Koszt pełnego skanu jest przy tym taki sam bez względu na to, do ilu kolumn chcemy się odwołać.

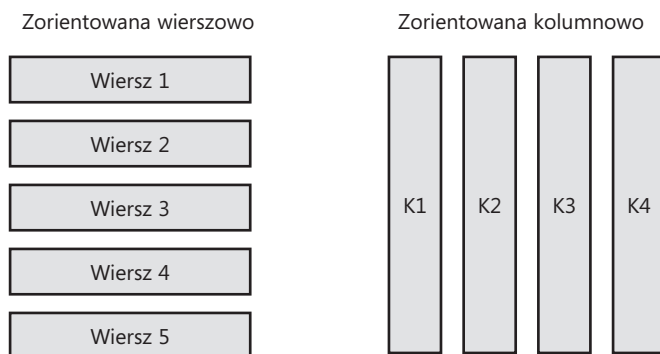
Zorientowana kolumnowo baza danych wykorzystuje inne podejście. Zamiast postrzegać wiersz tabeli jako podstawową jednostkę magazynowania, każda kolumna jest traktowana jako oddzielny byt i dane tej kolumny są przechowywane oddzielnie. Dla przykładu wyobraźmy sobie, że dane z naszej początkowej tabeli są logicznie przechowywane w następujący sposób:

Nazwa kolumny	Dane kolumny
ID_Author	1, 2, 3
FirstName	Alberto, Maurizio, Marco
LastName	Ferrari, Macagno, Russo
Blog	http://sqlblog.com/blogs/alberto_ferrari , http://adventureworks.com/blogs/mmacagno , http://sqlblog.com/blogs/marco_russo
Posts	27, 43, 38

Zapewnia to bardzo szybkie wykonanie zapytania o dane zawarte w jednej kolumnie, ale wymaga większego wysiłku obliczeniowego, jeśli zechcemy odczytać dane z wielu kolumn dla wybranego wiersza. Najgorszym przypadkiem jest tu, oczywiście, zażądanie danych ze wszystkich kolumn dla wybranego wiersza, co wymaga dostępu do miejsca magazynowania wszystkich wierszy. Na przykład w celu pobrania danych dotyczących Maurizio trzeba go zidentyfikować w kolumnie FirstName. Ponieważ jest to drugi element tej kolumny, pobranie drugiego elementu ze wszystkich pozostałych kolumn spowoduje zwrócenie wszystkich danych, które go dotyczą. Z drugiej jednak strony, aby wyliczyć całkowitą liczbę artykułów wysłanych przez wszystkich autorów, konieczne będzie odczytanie tylko zawartości kolumny Posts.

Z punktu widzenia odczytywania danych baza zorientowana kolumnowo może być szybsza, gdyż dostęp do informacji jest optymalny dla wielu scenariuszy zapytań. W istocie najczęstsze kwerendy w systemach analitycznych wymagają dostępu do danych zawartych w tylko nielicznych kolumnach, zazwyczaj w celu ich agregacji poprzez grupowanie wyników odpowiednio do wartości w innych kolumnach.

Rysunek 1-24 pokazuje porównanie przechowywania danych opartego na wierszach z opartym na kolumnach.



RYСУNEK 1-24 Bazy danych zorientowane wierszowo i kolumnowo.

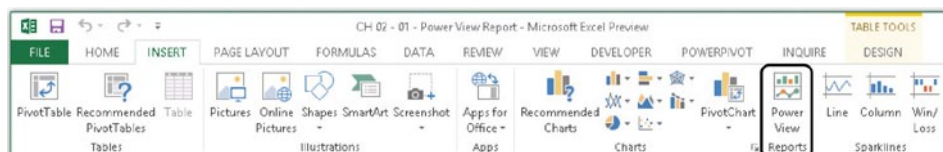
PowerPivot jest również bazą danych działającą w pamięci (*in-memory*). Oznacza to, że została ona zaprojektowana i zoptymalizowana przy założeniu, że cała baza danych jest załadowana do pamięci. Aby móc pomieścić więcej danych i poprawić wydajność, dane te są dodatkowo skompresowane i są dynamicznie dekompresowane podczas każdej kwerendy.

Tworzenie raportu Power View

Power View jest wsłaniałym narzêdziem do tworzenia raportów, które zostało zintegrowane z programem Excel 2013 jako dodatek. Podobnie jak PowerPivot, musi ono zostać uaktywnione przed pierwszym użyciem. Po aktywowaniu uzyskujemy opcję tworzenia raportów Power View opartych na modelu danych PowerPivot.

Power View nie działa z tabelami Exccla; wymaga umieszczenia danych w modelu danych. Tym samym, jeśli rozpoczniemy tworzenie raportu Power View, wszystkie tabele użyte w tym raporcie zostaną automatycznie dodane do modelu danych, tak samo jak w przypadku raportu tabeli przestawnej, jeśli wybierzemy więcej niż jedną tabelę Exccla jako źródło. Ponieważ Power View może działać tylko z modelem danych, przy tworzeniu raportu Power View nie mamy możliwości wyboru źródła danych – domyślnie jest to model danych. Przykład raportów Power View, które zbudujemy w tym podrozdziale, zawiera skoroszyt *CH01-03-Power View Report.xlsx*.

Aby utworzyć raport Power View (przy założeniu, że dodatek został już aktywowany), należy użyć przycisku **Power View**, dostępnego na karcie **WSTAWIANIE** (INSERT) wstążki programu Excel, pokazanego na rysunku 1-25.



RYСУNEK 1-25 Przycisk Power View umożliwia stworzenie nowego raportu Power View.

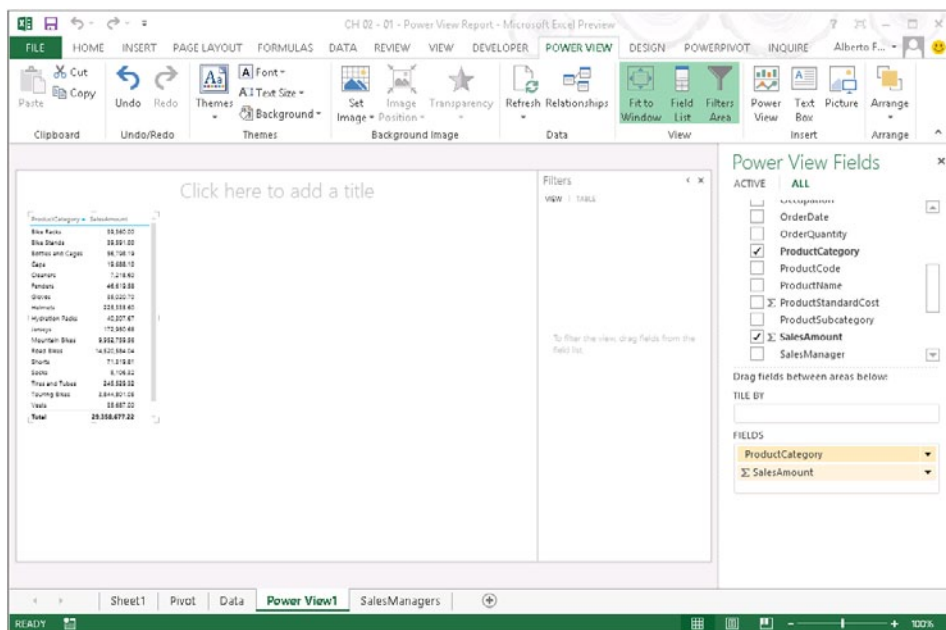
UWAGA Power View wymaga zainstalowania w systemie komponentu Silverlight. Jeśli składnik ten nie jest jeszcze zainstalowany, Power View wyświetli monit o jego instalację, przekierowując użytkownika do witryny, z której można pobrać odpowiednią wersję Silverlight. Po zainstalowaniu Silverlight dodatek Power View będzie działał poprawnie. Warto też zauważyć, że do pobierania właściwej wersji pliku instalacyjnego należy użyć przeglądarki Microsoft Internet Explorer, gdyż przy korzystaniu z innej przeglądarki trudne może być wybranie właściwej wersji Silverlight; wymaganej przez Exccla.



Utworzymy teraz prosty raport, wykorzystując ten sam skoroszyt, co w poprzednich częściach tego rozdziału. Środowisko Power View zostało zaprojektowane pod kątem prostoty: większość funkcji wymaga pojedynczego kliknięcia myszą, aby je uaktywnić i nie są dostępne wszystkie opcje konfiguracyjne, które znamy z klasycznych wykresów Exccla. Jednak główną przewagą Power View jest to, że pozwala utworzyć piękne raporty przy minimalnym wysiłku.

Po otwarciu raportu Power View ujrzymy pustą ramę raportu w lewej części okna oraz listę tabel zawartych w modelu danych po prawej. Lista ta odpowiada

liście pól, którą widzieliśmy wcześniej przy tworzeniu tabeli przestawnej. Tworzenie pierwszego raportu rozpoczniemy od rozwinięcia tabeli Sales i zaznaczenia kolumny ProductCategory. W rezultacie wewnątrz ramki pojawi się tabelka z listą kategorii. Po zaznaczeniu kolumny SalesAmount Power View doda wartość sprzedaży każdej kategorii i wyświetli ją w odpowiednim wierszu tabeli. W tym momencie okno programu Excel powinno wyglądać podobnie, jak na rysunku 1-26.



RYSUNEK 1-26 Raport Power View pokazujący kategorie i sprzedaż.

Koncepcja Power View polega na wyjściu od danych i dopiero późniejszym decydowaniu, jakiego formatu użyjemy dla nich. Załóżmy, że jesteśmy zainteresowani utworzeniem raportu pokazującego całkowitą sprzedaż dla każdej kategorii (co już w istocie zrobiliśmy), a poza tym chcemy zaprezentować sprzedaż w rozbiu geograficznym.

Pierwszy krok został już wykonany, zatem musimy teraz utworzyć nową tabelę, zawierającą kraje wraz z odpowiadającymi im wielkościami sprzedaży. W tym celu wystarczy kliknąć w pustym obszarze ramki Power View i wybrać najpierw kolumnę TerritoryCountry, a następnie SalesAmount. W tym momencie raport powinien wyglądać tak jak na rysunku 1-27.

ProductCategory	SalesAmount	TerritoryCountry	SalesAmount
Bike Racks	39.340.00	Australia	9.061.000.58
Bike Stands	39.591.00	Canada	1.977.844.06
Bottles and Cages	56.796.19	France	2.644.017.71
Caps	19.688.10	Germany	2.694.312.34
Cleaners	7.216.60	United Kingdom	3.391.712.21
Fenders	46.019.58	United States	9.369.789.51
Gloves	35.020.70	Total	29.358.677.22
Helmets	225.335.60		
Hydration Packs	40.307.67		
Jerseys	172.950.68		
Mountain Bikes	9.952.759.56		
Road Bikes	14.520.584.04		
Shorts	71.319.81		
Socks	5.106.32		
Tires and Tubes	245.529.32		
Touring Bikes	3.844.801.05		
Vests	35.687.00		
Total	29.358.677.22		

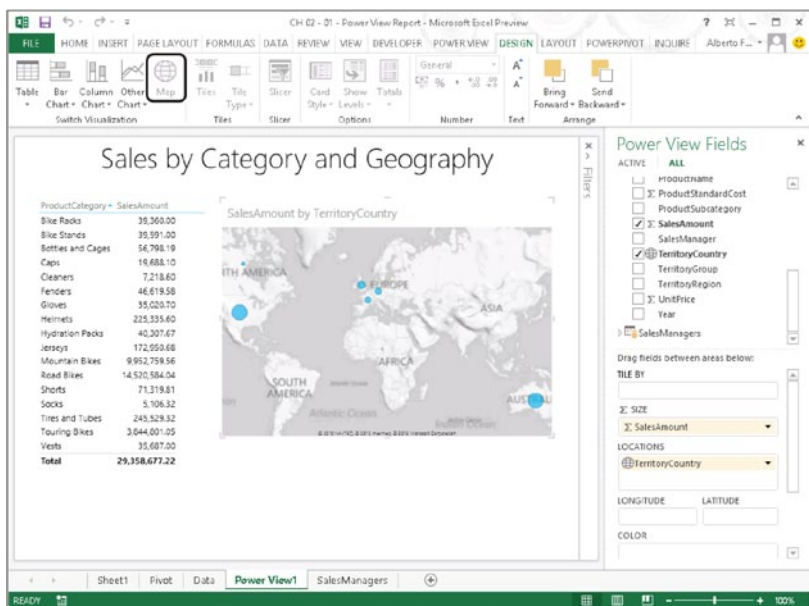
RYСУNEK 1-27 Raport Power View domyślnie wyświetla dane w postaci tabel.

Choć mamy już potrzebne dane, nie wygląda to zbyt interesująco. Tabelaryczne przedstawianie informacji jest zdecydowanie mniej niż piękne – potrzebujemy sposobu zademonstrowania tych informacji w formie wykresu. Zaczniemy od danych geograficznych. Byłoby znacznie lepiej, gdybyśmy mogli zaprezentować wielkość sprzedaży na mapie. Aby wykonać tę operację, wystarczy zaznaczyć tabelę zawierającą dane *TerritoryCountry* i użyć przycisku **Mapa** (Map) na karcie **PROJEKT** (DESIGN) wstążki Excela, jak na rysunku 1-28. Tabela zostanie błyskawicznie zastąpiona mapą świata, na której wielkości sprzedaży zostaną zobrazowane przez wielkość kółek umieszczonych na poszczególnych krajach.

WSKAZÓWKA Mapa do działania wymaga aktywnego połączenia internetowego, gdyż zarówno sam obraz mapy, jak i geograficzne rozpoznawanie krajów jest zapewniane przez usługę Bing³.



³ Polscy czytelnicy (a ogólniej – osoby z krajów nieanglojęzycznych, jak Polska) mogą zadawać sobie pytanie, czy mechanizm ten będzie działał, jeśli nazwy krajów zostaną wpisane w innym języku (np. w polskim). Odpowiedź brzmi – tak, będzie działać (przynajmniej dla bardziej popularnych języków). Trzeba jednak uważać, aby zachować konsekwencję – możemy używać nazwy „Germany” lub „Niemcy” i uzyskamy wizualizację danych we właściwym miejscu, ale jeśli w naszych danych wystąpią obie wersje nazwy, w efekcie otrzymamy dla Niemiec dwa kołka koncentryczne, o rozmiarach odpowiadających obu wpisom. Inaczej mówiąc – usługa Bing wie, że „Niemcy” to „Germany”, ale model danych PowerPivot, a w konsekwencji Power View, o tym nie wie.

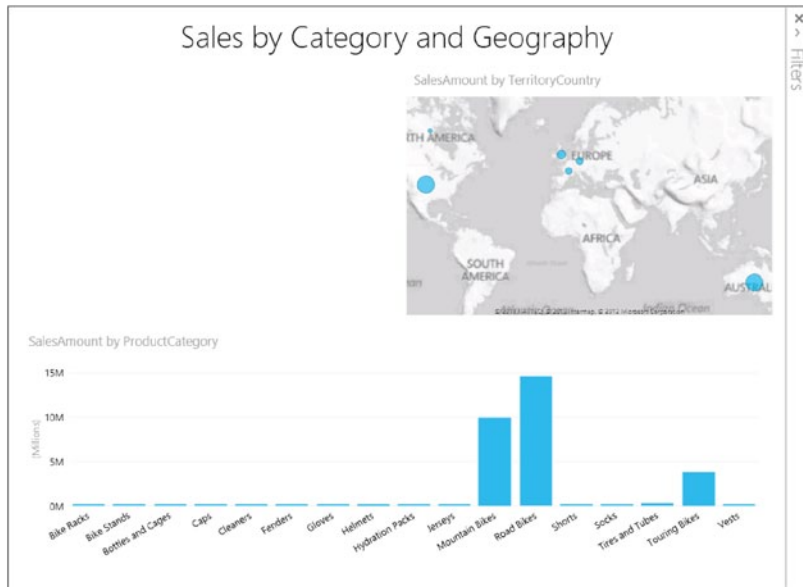


RYSUNEK 1-28 Przycisk Mapa przekształca tabelaryczne przedstawienie danych na ich reprezentację graficzną na mapie.

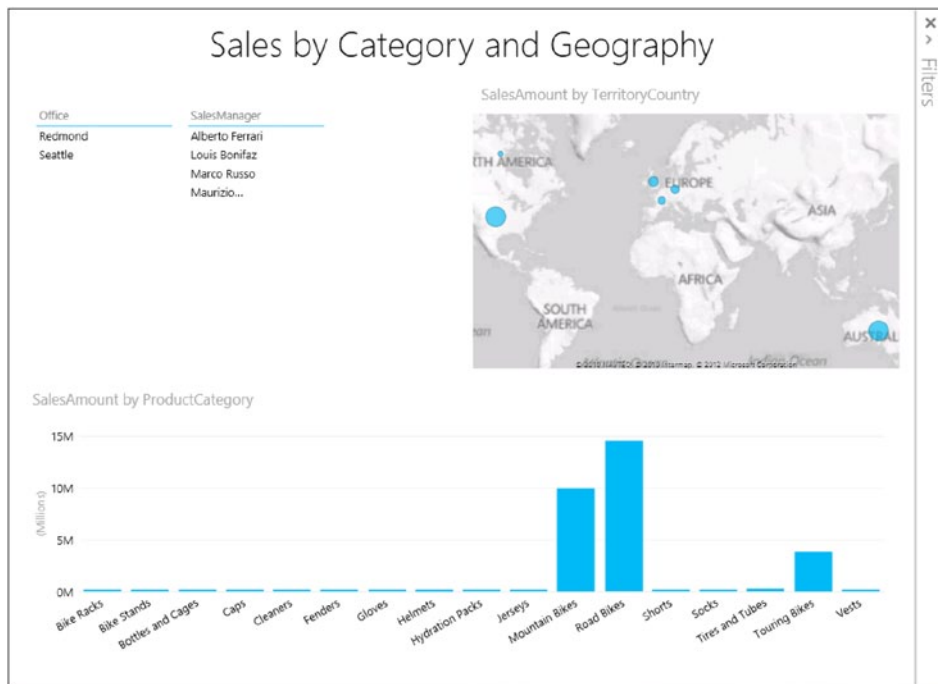
Teraz możemy wykonać podobną procedurę, aby przetworzyć tabelę Categories w wykres kolumnowy, skorygować jego rozmiary i umieścić go poniżej mapy. Jak można zauważyć, koncepcję działania Power View można podsumować następująco: *najpierw dane, potem reprezentacja graficzna*. Nasz wykres w tym momencie powinien wyglądać podobnie do pokazanego narysunku 1-29.

Teraz można kliknąć w pustym miejscu raportu i, wykonując te same kroki, dodać jeszcze dwie tabele: jedną zawierającą kolumnę Office z tabeli SalesManagers oraz drugą z kolumną SalesManager z tej samej tabeli. Jak już wspomnieliśmy, konieczne jest kliknięcie pustego miejsca raportu, aby utworzyć nową tabelę. Gdyby wybrać nowe pole przy zaznaczonej tabeli, zostałoby ono dołączone (jako kolumna) do tej tabeli.

Raport w tym momencie zawiera dwie tabele, jedną mapę i jeden wykres kolumnowy, jak na rysunku 1-30.



RYSUNEK 1-29 Tabela Categories przekształcona w wykres kolumnowy

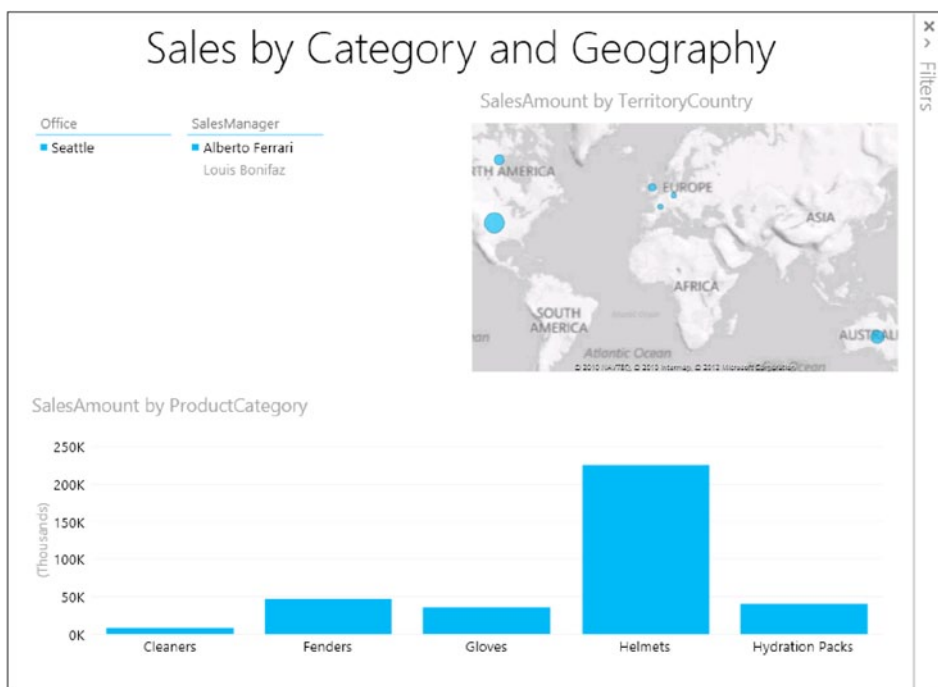


RYSUNEK 1-30 Raport po dodaniu tabel Office i SalesManager



UWAGA Teraz, skoro już zaczęliśmy tego używać, warto krótko wyjaśnić, czym jest, a czym nie jest Power View. Jest to narzędzie do graficznej eksploracji danych, zaprojektowanym w celu śledzenia i analizowania informacji poprzez wykresy, a nie tabele przestawne (złożone głównie z liczb). W ten sposób wystarczy spojrzeć na obraz, aby odkryć coś, co jest warte bliższego zbadania. Następnie możemy wrócić do tabeli przestawnej i drążyć informacje, aby poznać szczegóły. Power View *nie jest* natomiast środowiskiem do tworzenia wyrafinowanych wykresów i nie ma na celu zastąpić specjalistycznego oprogramowania tego typu. Jego celem jest umożliwienie bardzo szybkiego i możliwie prostego tworzenia wykresów i przeglądanie danych w różnych formatach.

Na koniec możemy przekształcić dwie ostatnie tabele na fragmentatory, używając przycisku **Fragmentator** (Slicer) na karcie **PROJEKT** wstążki Excela. Przekonwertowanie tabeli na fragmentator pozwala wykorzystać ją do filtrowania raportu. Po kliknięciu wiersza we fragmentatorze cały raport będzie prezentował dane przefiltrowane na podstawie tego wyboru. Na rysunku 1-31 pokazany jest gotowy raport po wybraniu biura Seattle i Alberto Ferrari jako menedżera.



RYСУNEK 1-31 Raport podobny do powyższego można utworzyć w kilkanaście sekund.

Opisanie w książce wrażenia, jakie wywołuje wykonywanie tych działań, jest niemożliwe, ale jesteśmy pewni, że prostota tych operacji i łatwość, z jaką utworzyliśmy ten raport, mówią same za siebie.

Tworzenie pięknych graficznych zobrazowań danych w Power View jest naprawdę proste. Musimy jednak pamiętać, że prostota jest najważniejszą cechą Power View. Nie warto tracić czasu na szukanie tu bogactwa i złożoności wykresów dostępnych w Excelu. Mechanizm wykresów Excela pozwala utworzyć dowolny rodzaj wykresu, udostępniając tysiące (to nie przesada) rozmaitych opcji i ustawień. Power View to wybór drogi w przeciwnym kierunku: kilka kliknięć wystarczy do utworzenia interesującego wykresu, ale jeśli potrzebne są większe możliwości konfiguracji, wówczas Power View nie jest tym narzędziem, którego należy użyć.

Aby zapewnić tę niezwykłą łatwość użycia, Power View wymaga, aby model danych zawierał pewne informacje o treści naszych danych i jak należy je najlepiej pokazywać. Więcej informacji na temat Power View i jego interakcji z modelem danych znajduje się w rozdziale 10, „Korzystanie z Power View”.

ROZDZIAŁ 2

Korzystanie z unikatowych funkcji PowerPivot

W poprzednim rozdziale przedstawiliśmy podstawowe informacje na temat *modelu danych* w programie Microsoft Excel 2013. Można było zauważyć, że kluczem do wykorzystania najbardziej interesujących nowych funkcji programu Excel 2013 jest utworzenie modelu danych PowerPivot. Na koniec pokazaliśmy, że silnik PowerPivot działa nawet wtedy, gdy sam dodatek PowerPivot nie został aktywowany, gdyż silnik analityczny xVelocity został zintegrowany z Excelem i rozpoczyna działanie, gdy tylko użytkownik utworzy tabelę przestawną odwołującą się do więcej niż jednej tabeli danych albo zacznie tworzyć raport Power View.

Czytelnik mógłby w tym momencie zadać pytanie, dlaczego w ogóle konieczne jest włączanie (aktywowanie) dodatku, skoro silnik PowerPivot może zostać uruchomiony i wykorzystany bez tego. Przyczyna leży w tym, że Excel domyślnie udostępnia jedynie małą część funkcjonalności PowerPivot. Większość z zaawansowanych, bardziej interesujących funkcji i cech dostępnych jest jedynie poprzez sam dodatek. W tym rozdziale wstępnie zapoznamy się z tymi funkcjami. Każda z nich zostanie szczegółowo przeanalizowana i omówiona w dalszych rozdziałach książki. W tym rozdziale mamy na celu przedstawić widok ogólny najważniejszych funkcji PowerPivot, które są dostępne tylko poprzez dodatek, a nie inne obszary Excela.

Ładowanie danych ze źródeł zewnętrznych

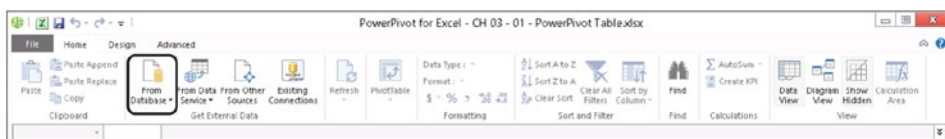
W przykładach pokazanych w pierwszym rozdziale łądowniliśmy dane do skoroszytu Excel i następnie, odwołując się do tych danych (lokalnych dla skoroszytu), tworzyliśmy model danych PowerPivot. Przypomnijmy, że PowerPivot przechowuje swoje dane oddzielnie od danych arkuszy Excela. Oznacza to, że tworząc model danych w ten sposób, w praktyce zużywamy dwukrotnie więcej miejsca na nasze dane.

W istocie jest to nieco mniej niż dwa razy, gdyż model danych PowerPivot opiera się na mechanizmie xVelocity, który silnie kompresuje dane załadowane do pamięci (i zapisane w pliku skoroszytu), podczas gdy Excel nie kompresuje danych w swoich tabelach i używa mniej wydajnej techniki kompresji, gdy dane są zapisywane na dysku. Co więcej, PowerPivot jest w stanie przechować w pamięci setki milionów wierszy danych, podczas gdy Excel ciągle jest ograniczony do (około) jednego miliona wierszy na tabelę. Tak więc ładowanie danych najpierw do Excela, a dopiero potem do modelu

danych nie jest dobrym wykorzystaniem zasobów: granicę możliwości Excela napotkamy długo przed tym, zanim wyczerpią się możliwości PowerPivot.

Jedną z najbardziej interesujących cech PowerPivot jest jego zdolność do ładowania danych bezpośrednio do modelu danych, przy czym Excel w ogóle nie musi o tym wiedzieć. W ten sposób ładujemy dane tylko raz i w najlepszym formacie (silnie skompresowane). Aby dać chociaż przybliżone wyobrażenie, co to oznacza, zauważmy, że tabela sprzedaży używana w rozdziale 1, „Wprowadzenie do PowerPivot” do analizy dała w rezultacie plik skoroszytu o wielkości około 10 MB. Ta sama tabela, ale załadowana bezpośrednio do PowerPivot bez pośredniego kroku tworzenia arkusza Excela daje się zapisać w skoroszytcie, który ma wielkość tylko 1 MB (tak, 10 razy mniejsza).

W celu załadowania danych bezpośrednio do bazy PowerPivot (modelu danych) musimy otworzyć okno PowerPivot, wybierając kartę **POWERPIVOT** we wstążce i klikając przycisk **Zarządzaj** (Manage). Po otwarciu okna PowerPivot możemy załadować dane, używając przycisku **Z bazy danych** (From Database) na karcie **Narzędzia główne** (Home) wstążki PowerPivot, jak na rysunku 2-1.



RYСУNEK 2-1 PowerPivot udostępnia opcję załadowania danych bezpośrednio z bazy do modelu danych.

Do ładowania danych z różnych baz można wykorzystać wiele odmiennych sterowników, co omówimy szczegółowo w rozdziale 6, „Ładowanie danych”. Na potrzeby tej małej demonstracji wykorzystamy połączenie z Microsoft SQL Server. Tym samym z listy rozwijanej możliwych baz danych należy wybrać opcję **Z programu SQL Server** (From SQL Server).

W tym momencie PowerPivot uruchamia **Kreator importu tabeli** (Table Import Wizard), który prowadzi użytkownika przez kolejne kroki procesu ładowania danych. Pierwsza strona kreatora (rysunek 2-2) żąda podania parametrów połączenia: trzeba wyspecyfikować nazwę serwera utrzymującego bazę danych (w tym przykładzie Demo) oraz nazwę bazy (AdventureWorks DW2012). Skoroszyt z już załadowanymi danymi jest dostępny w plikach przykładowych jako *CH02-01-PowerPivot Data Model.xlsx*.

Po kliknięciu **Dalej** (Next) przechodzimy do drugiej strony kreatora, gdzie musimy określić, czy zamierzamy ładować wybrane tabele (w całości), czy też użyć kwerendy, co demonstuje rysunek 2-3.

Table Import Wizard

Connect to a Microsoft SQL Server Database
Enter the information required to connect to the Microsoft SQL Server database.

Friendly connection name:

Server name:

Log on to the server:

☒ Use Windows Authentication

☐ Use SQL Server Authentication

User name:

Password:

☐ Save my password

Database name:

RYSUNEK 2-2 Kreator importu tabeli wymaga podania podstawowych informacji potrzebnych do wykonania ładowania danych.

Table Import Wizard

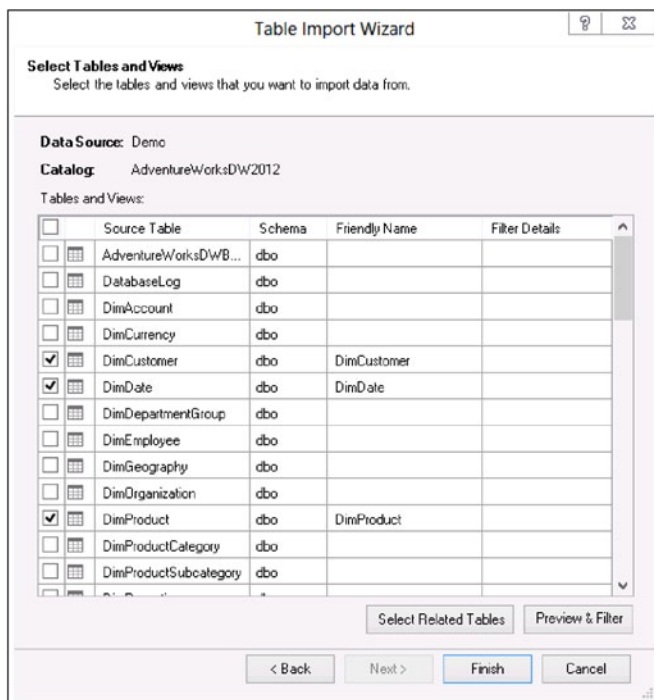
Choose How to Import the Data
You can either import all of the data from tables or views that you specify, or you can write a query using SQL that specifies the data to import.

☒ Select from a list of tables and views to choose the data to import

☐ Write a query that will specify the data to import

RYSUNEK 2-3 Możemy ładować dane z wybranych tabel lub napisać zapytanie w języku SQL.

Ładowanie z kwerendy jest nieco bardziej skomplikowane, gdyż wymaga znajomości języka SQL. Dlatego w tym prostym przykładzie wybierzemy opcję **Dokonaj wyboru z listy tabel i widoków**, aby wybrać dane do zaimportowania (Select From A List Of Tables And Views To Choose The Data To Import). W rezultacie kreator wyświetli listę wszystkich tabel dostępnych w wybranej bazie danych, jak na rysunku 2-4.



RYСУNEK 2-4 Aby załadować tabelę, trzeba po prostu zaznaczyć pole wyboru przy jej nazwie.

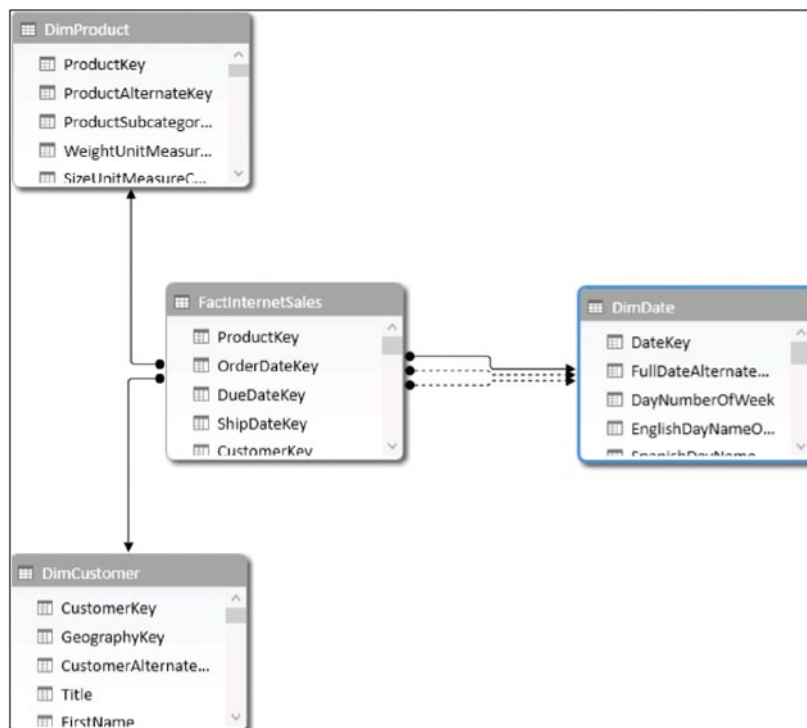
Po kliknięciu **Zakończ** (Finish) PowerPivot rozpocznie ładowanie do pamięci danych z wybranych tabel. Po ukończeniu wyświetli raport podsumowujący i ponownie wyświetli okno PowerPivot, które teraz prezentuje wszystkie dane załadowane do modelu danych.

Trzeba tu zauważyć, że gdy PowerPivot łączy tabele z serwera, nie ogranicza się do prostego skopiowania zawartych w nich informacji, ale wykonuje także ważny krok dodatkowy: analizuje istnienie relacji w bazie danych i jeśli taka relacja zostanie wykryta i może zostać przeniesiona do PowerPivot, zostanie automatycznie utworzona.



UWAGA Istnieją sytuacje, w których PowerPivot nie może załadować relacji do swojego modelu danych albo gdy relacja może zostać załadowana, ale nie jest uaktywniona. Omówienie takich przypadków wykracza poza zakres tego rozdziału. Badaniem tego zagadnienia zajmiemy się w rozdziale 6.

W tym przykładzie wybraliśmy do załadowania cztery tabele: DimProduct, DimCustomer, FactInternetSales oraz DimDate. Jeśli przełączymy się do widoku Diagramu zaraz po załadowaniu tych tabel, co pokazuje rysunek 2-5, można będzie zauważyć, że kilka relacji zostało automatycznie utworzonych w procedurze ładowania, gdyż PowerPivot wykrył ich istnienie w źródłowej bazie danych.



RYСУNEK 2-5 Relacje są automatycznie ładowane do modelu danych, jeśli istnieją w źródłowej bazie danych.

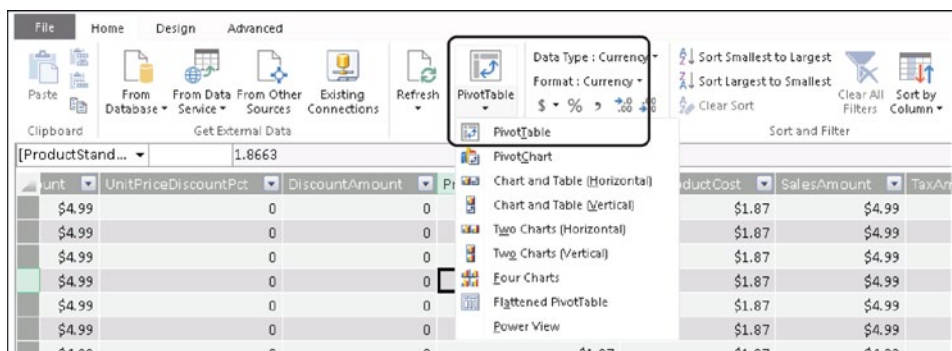
Jak można zauważyć, ładowanie danych bezpośrednio do modelu danych PowerPivot jest proste i naturalne – to zadziwiające, że jest nawet prostsze od ładowania danych do tabeli Excel. Trzeba jednak zauważyć, że pominięcie dodatkowego kroku ładowania danych do tabeli Excel pociąga za sobą pewną niedogodność: tabele PowerPivot są tylko do odczytu. Warto dwukrotnie przeczytać ostatnie zdanie i dobrze je sobie utrwalić. Jest to bowiem wielkie ograniczenie, które należy zawsze mieć na uwadze przy podejmowaniu decyzji, czy załadować dane do arkusza Excela i następnie do PowerPivot, czy pominąć ten pierwszy krok.

Jeśli załadujemy tabele do Excela, mamy do dyspozycji wszystkie funkcje edycyjne, jakimi dysponuje program. Jeśli tabela zostanie załadowana wprost do modelu danych PowerPivot, żadna edycja nie jest dozwolona. Na przykład nie można usunąć zbędnej

kolumny, jak w tabeli Excela. Nie oznacza to jednak, że nie można dokonywać zmian. W istocie, przy korzystaniu z PowerPivot mamy do dyspozycji możliwość wykonywania najrozmaitszych obliczeń na danych w tabelach, a ponadto można dołączyć do oryginalnych tabel nowe kolumny, posługując się językiem DAX. Jednak w żadnym przypadku nie można będzie zmienić oryginalnej zawartości. W praktyce ta właściwość może być również postrzegana jako zaleta: chroni przed przypadkowym lub pomyłkowym zmodyfikowaniem czy zamazaniem danych. Jednak bez względu na to, jak na to spojrzymy, jest to cecha, którą trzeba wziąć pod uwagę przy tworzeniu rozwiązania.

Tworzenie tabeli przestawnej PowerPivot

Tabelę przestawną opartą na modelu danych PowerPivot można utworzyć bezpośrednio w oknie PowerPivot, klikając przycisk **Tabela przestawna** (PivotTable) na karcie **Narzędzia główne** wstążki PowerPivot, jak na rysunku 2-6.



RYСУNEK 2-6 Tabelę przestawną można utworzyć bezpośrednio w oknie PowerPivot.

Okno dialogowe wyświetlane po wybraniu tego polecenia jest bardzo proste – znacznie prostsze niż to używane w Excelu (rysunek 2-7). Powód jest prosty – tabela przestawna tworzona z okna PowerPivot jest domyślnie tabelą opartą na modelu danych PowerPivot. Tak więc nie ma potrzeby pytania użytkownika o źródło danych czy dowolny inny parametr: Excel wie już, że będziemy chcieli odpytywać model danych PowerPivot.

Kliknięcie OK tworzy nową tabelę przestawną w nowym arkuszu. Można teraz wybrać odpowiednie pola. W naszym przykładzie umieściliśmy pole **Color** z tabeli **DimProduct** w obszarze wierszy, pole **FiscalYear** z tabeli **DimDate** w obszarze kolumn, a pole **SalesAmount** z tabeli **FactInternetSales** w obszarze wartości. Wynikowa tabela przestawna jest pokazana na rysunku 2-8.

Interesujące, że przy minimalnym wysiłku byliśmy w stanie zagregować i podzielić dane pochodzące z trzech różnych tabel i uzyskać ciekawy raport. Magia, która odbywa się w tle, wynika z istnienia relacji pomiędzy tabelami: to dzięki nim mogliśmy