

Business Intelligence mit Excel

Datenanalyse und Reporting mit PowerQuery, PowerPivot und PowerBI Desktop. Für Excel 2010 bis 2019

Bearbeitet von
Von Ignatz Schels

1. Auflage 2019. Buch. 501 S. Inkl. eBook. Softcover

ISBN 978 3 446 45711 9

Format (B x L): 17,4 x 24,4 cm

Gewicht: 1087 g

[Weitere Fachgebiete > EDV, Informatik > Datenbanken, Informationssicherheit, Geschäftssoftware > Automatische Datenerfassung, Datenanalyse](#)

schnell und portofrei erhältlich bei


DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung beack-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.

HANSER



Leseprobe

zu

Business Intelligence mit Excel

von Ignatz Schels

ISBN (Buch): 978-3-446-45711-9

ISBN (E-Book): 978-3-446-45756-0

Weitere Informationen und Bestellungen unter

<https://www.hanser-fachbuch.de/buch/Business+Intelligence+mit+Excel/9783446457119>

sowie im Buchhandel

© Carl Hanser Verlag, München

Inhalt

Vorwort	XI
1 Einleitung	1
1.1 Excel im Business Intelligence – eine Chance	2
1.2 Werkzeuge und Methoden	3
1.2.1 Power Query	4
1.2.2 Power Pivot	5
1.2.3 Power BI	6
1.2.4 Power View	8
1.2.5 Power BI oder Excel?	8
1.2.6 Power Pivot oder Datenbank?	9
2 Grundlagen, Verfahren und Konzepte	11
2.1 Relationale und multidimensionale Datenbanken	11
2.2 Das Entity-Relationship-Modell (ERM)	12
2.3 Normalisierung und Faktentabellen	12
2.4 OLAP – die multidimensionale Datenbank	14
2.4.1 Von OLAP zu BI	14
2.4.2 Wie ein OLAP-Cube funktioniert	14
2.5 Datenimport und dynamische Verknüpfungen	19
2.5.1 ODBC und MS Query	19
2.5.2 ODBC-Treiber verwalten	19
2.6 SQL	20
2.6.1 SQL-Abfragen mit Excel	21
3 Das Datenmodell	33
3.1 Regeln für Datenmodelle	33
3.2 Spezifikationen und Einschränkungen	34
3.2.1 Objekte im Datenmodell	34
3.3 Datenmodell anlegen	35
3.3.1 Über die PivotTable	35

3.3.2	Als Tabelle über Power Pivot	37
3.3.3	Implizites und explizites Datenmodell	38
3.3.4	Datenmodell entfernen	42
3.3.5	Datenmodelle zusammenführen	43
3.4	Datenmodelle komprimieren und Speicherplatz sparen	43
3.4.1	Nur importieren, was nötig ist	44
3.4.2	Weniger eindeutige Werte	44
3.4.3	Nur die benötigten Tabellen aus der Datenbank	44
3.4.4	Spaltenzahl reduzieren	45
3.5	Optionen für den Datenimport	48
4	Abrufen und Transformieren (Power Query)	49
4.1	Menüband und Symbole	49
4.2	ETL – der Prozess	50
4.3	Power Query installieren	52
4.4	Daten abrufen und transformieren	53
4.4.1	Aus Datei	53
4.4.2	Aus Datenbank	54
4.4.3	Aus Azure	54
4.4.4	Aus Online-Diensten	55
4.4.5	Aus anderen Quellen	55
4.5	Abfrageoptionen	58
4.5.1	Daten laden	58
4.5.2	Abfrage-Editor	58
4.5.3	Sicherheit	59
4.5.4	Datenschutz	59
4.5.5	Regionale Einstellungen	59
4.5.6	Konto und Ablaufverfolgung	60
4.5.7	Aktuelle Arbeitsmappe	60
4.6	Abfrage erstellen	61
4.6.1	Eine neue Abfrage	62
4.6.2	Abfragetools/Abfrage bearbeiten	68
4.7	Der Power Query-Editor	69
4.7.1	Die Elemente	70
4.7.2	Abfragesprache einstellen	70
4.7.3	Abfrageeigenschaften	70
4.7.4	Abfrage abbrechen	71
4.7.5	Abfrage speichern	71
4.7.6	Transformationen anwenden	72
4.7.7	Gruppieren	78
4.7.8	Daten transformieren und entpivotieren	82
4.7.9	Abfrageschritte bearbeiten	85
4.7.10	Schritte benennen	92

4.7.11	Abfragen kombinieren	92
4.7.12	Abfragen anfügen	98
4.8	Abfragen kopieren und transferieren	101
4.8.1	M-Code transferieren	101
4.8.2	Abfrage kopieren	102
4.8.3	Abfragen als ODC-Datei speichern	105
4.8.4	Abfrage an den Azure-Datenkatalog senden	107
4.9	Berechnete Spalten	110
4.9.1	Das Prinzip	110
4.10	Mit Parametern arbeiten	117
4.10.1	Abfrage mit Parameter auf Pfad und Dateiname	122
4.10.2	Abfrage verfeinern mit M-Code	131
4.10.3	Parameterabfrage mit M-Funktion	133
4.10.4	Parameterabfrage mit Funktion und WENN-Filter	133
4.11	Die Abfragesprache M	135
4.11.1	Die Bearbeitungsleiste	136
4.11.2	Der erweiterte Editor	137
4.11.3	Die Syntax der M-Formel	139
4.11.4	Funktionsreferenzen und Hilfe	146
4.11.5	Abfrage auf Funktionsliste erstellen	146
4.11.6	Datumsberechnungen mit M	147
4.11.7	Kalendertabelle aus Start-/Endedatum erstellen	153
4.12	Power Query-Praxis	156
4.12.1	Fehlzeitenanalyse	156
4.12.2	Richtig filtern	163
4.12.3	Webabfrage mit Formularelement kombiniert	168
4.12.4	Vorsicht bei automatischer Typerkennung	177
4.12.5	Abfragen kombinieren mit Joins	177
4.12.6	Zellwerte in Abfragen verwenden	185
4.12.7	Daten aus Zellen im Kopfbereich übernehmen	190
4.12.8	Daten mit variablen Spaltenüberschriften	196
4.12.9	Dateien aus Ordnern	198
4.12.10	Zuletzt gespeicherte Datei importieren	202
4.12.11	Quelldaten mit unterschiedlichen Spaltenüberschriften	205
4.12.12	SAP-Berichte	211
5	Power Pivot – Daten modellieren im Datenmodell	219
5.1	Grundlagen	219
5.1.1	Das kann Power Pivot	219
5.1.2	Technische Voraussetzungen	220
5.1.3	Office-Version	220
5.2	Power Pivot starten	221
5.2.1	Zurück zu Excel	221

5.3	Die Power Pivot-Menüs	222
5.3.1	Das Datei-Menü	222
5.3.2	Das Menü „Start“	223
5.3.3	Das Menü „Entwurf“	233
5.3.4	Das Menü „Erweitert“	237
5.4	Daten importieren	241
5.4.1	Zwischenablage	241
5.4.2	Aus Datenbank – Aus SQL Server	243
5.4.3	Aus Datenbank – Aus Access	251
5.4.4	Aus Datenbank – Analysis Services	255
5.4.5	Aus OLAP-Cubes per MDX	255
5.4.6	Aus Datendienst	260
5.4.7	Aus Azure SQL	266
5.4.8	Aus Excel	276
5.4.9	Aus Text- oder CSV	279
5.4.10	Aus anderen Quellen	282
5.4.11	SQL-Abfragen	282
5.5	Tabelleneigenschaften	285
5.5.1	Datumstabelle	286
5.5.2	Die Tabellenregister	307
5.6	Arbeiten mit Spalten	309
5.6.1	Spalte einfügen	310
5.6.2	Spalten formatieren	312
5.6.3	Spalte löschen	314
5.6.4	Beschreibungen für Spalten und Measures	316
5.7	Hierarchien	316
5.7.1	Hierarchien erstellen	317
5.7.2	Hierarchien in PivotTables verwenden	320
5.8	Measures erstellen und verwalten	324
5.8.1	Implizite Measures	324
5.8.2	Diskrete Anzahl	329
5.8.3	Explizite Measures	332
5.9	Time-Intelligence und KPIs	339
5.10	PivotTable in Cube-Funktionen umwandeln	344
5.10.1	Cube-Funktionen-Übersicht	344
5.10.2	Beispiel: Firmenumsätze	345
5.11	Einführung in DAX	350
5.11.1	Was ist DAX?	351
5.11.2	DAX-Funktionen erstellen	352
5.11.3	DAX-Funktionen im Kontext	357
5.11.4	Die wichtigsten DAX-Funktionen	364
5.11.5	Hilfe zu DAX-Funktionen	366

6	Karten und 3D-Karten	369
6.1	Kartendiagramme	369
	Beispiel: Einwohner pro Bundesland	370
	Bundesländer mit Kategorie-Einfärbung	371
6.2	3D-Karten	373
7	BI-Analysen mit PivotTables und PivotCharts	377
7.1	Das Pivot-Prinzip	379
7.1.1	Von der Filtersumme zur PivotTable	379
7.1.2	Voraussetzungen	382
7.1.3	Auswertbare Daten	383
7.2	Datenquellen für PivotTables	385
7.2.1	Listen	385
7.2.2	Tabellen manuell erstellen	387
7.2.3	MS Query	389
7.2.4	Power Query	390
7.2.5	Power Pivot	393
7.3	PivotTable-Berichte erstellen	394
7.3.1	Berichte planen	394
7.3.2	PivotTables positionieren	395
7.4	Das Berichtslayout	398
7.4.1	Die Feldliste	399
7.4.2	Layouts und Formate	400
7.4.3	Layout als Standard speichern	403
7.4.4	Wertfeldeinstellungen	404
7.4.5	Zahlenformate	406
7.4.6	Bedingte Formatierung	406
7.4.7	PivotTables aktualisieren	407
7.4.8	Datenquelle ändern	410
7.4.9	PivotTable verschieben und löschen	410
7.4.10	Drilldown (Details anzeigen)	411
7.5	Datenschnitte	412
7.5.1	Ein neuer Datenschnitt	412
7.5.2	Datenschnitt formatieren	413
7.5.3	Datenschnittverbindungen	414
7.6	Die Zeitachse	415
7.6.1	Eine neue Zeitachse	415
7.6.2	Zeitachse bedienen	415
7.7	Berechnete Felder	416
7.7.1	Berechnete Felder und Elemente in der PivotTable	416
7.7.2	Berechnete Felder mit strukturierten Verweisen	420
7.7.3	Berechnete Felder mit Power Query	423
7.7.4	Berechnete Felder mit DAX in Power Pivot	428

7.8	Gruppieren	432
7.8.1	Textfeld gruppieren	432
7.8.2	Wertefeld gruppieren	435
7.8.3	Datumfelder gruppieren	435
7.9	PivotCharts	437
7.9.1	Das Prinzip	438
7.9.2	Diagrammtypen	441
7.9.3	Beispiel: Altersstrukturanalyse	442
7.9.4	Kennzahlencockpits mit PivotCharts	448
8	Power BI	455
8.1	Das Prinzip	456
8.2	Power BI und Excel	457
8.3	Power BI Desktop	458
8.3.1	Komponenten und Ansichten	459
8.3.2	Daten importieren	461
8.3.3	Daten modellieren mit Power Query	461
8.3.4	Eine Datumstabelle mit DAX	464
8.3.5	Beziehungen und Measures mit Power Pivot	466
8.3.6	Visualisierungen	467
8.3.7	Interaktive Filter	473
8.4	Mit Power BI Services in der Cloud	474
8.4.1	Bericht veröffentlichen	474
8.5	Von Excel zu Power BI	476
8.5.1	Excel-Arbeitsmappe veröffentlichen	477
8.5.2	Excel-Daten exportieren	479
8.5.3	Power BI Publisher	481
Index	485	

Vorwort

Liebe Leserin, lieber Leser,

ich freue mich, dass Sie sich entschlossen haben, mit diesem Buch in das Thema „Business Intelligence mit Excel“ einzusteigen. Vielleicht haben Sie ja schon erste Versuche mit BI-Tools gemacht und wollen Ihr Wissen jetzt vertiefen. Oder Sie gehören zu den klassischen Excel-Anwendern und suchen nach Lösungen, um Ihre Arbeit zu optimieren. Es kann auch sein, dass Sie den Auftrag bekommen haben, sich im Markt nach einer geeigneten Lösung für ein BI-basiertes Reporting umzusehen, und jetzt wollen Sie wissen, ob das auch mit Excel machbar ist und wo die Grenzen sind.

In jedem Fall sind Sie richtig beraten, sich mit den BI-Tools von Excel (und mit diesem Buch) zu beschäftigen. Die Tabellenkalkulation ist nach wie vor das wichtigste Werkzeug für Datenverarbeiter am PC und hat mit Business Intelligence massiv an Bedeutung gewonnen. Neue Tools sind an Bord – Power Query, Power Pivot, Power BI –, behalten Sie den Überblick und informieren Sie sich mit diesem Buch über Vor- und Nachteile, Einsatz und Nutzen der einzelnen Werkzeuge.

Für den praktischen Nutzen ist gesorgt. Wir sind seit Jahren zum Thema *Business Intelligence mit Excel* in Seminaren, Inhouse-Schulungen und Workshops unterwegs und haben bereits mehrere Projekte auf Basis der BI-Tools erfolgreich abgeschlossen. Schauen Sie auf unserer Webseite vorbei und schreiben Sie uns. Wir beraten Sie gerne:



www.schels.de

Wir – das sind Ignatz Schels und Ignaz Schels junior. Von Ignaz Schels junior stammt das Buch zur neuen Programmiersprache für die BI-Tools,

M – Daten abfragen und verarbeiten mit Excel und Power BI,

das ebenfalls im Hanser-Verlag erschienen ist (ISBN 978-3-446-45588-7).

Die Beispiele im Buch

... sind praxisnah und mit viel Erfahrung ausgearbeitet. Zu den wichtigsten Themen finden Sie immer eine oder mehrere Übungen mit Schritt-für-Schritt-Anleitungen, und um diese nachzuvollziehen, brauchen Sie Beispieldaten. Diese stehen auf der Webseite des Verlags zum Download bereit:

<https://downloads.hanser.de>

Zu jeder Übung finden Sie eine von externen Verknüpfungen freie Basisdatei sowie eine Lösungsdatei. Da diese in der Regel Verbindungen zu Datenquellen enthält, beziehen sich alle Verknüpfungen auf ein einheitliches Verzeichnis. Kopieren Sie deshalb alle Beispieldaten in diesen Ordner auf Ihrem Root-Verzeichnis (C:), dann funktionieren auch die Lösungen problemlos:

C:\HanserExcelBI

Beispieldateien haben beliebige Bezeichnungen wie *Umsatz2018.csv* oder *Getränkemarkt.xlsx*. Lösungen beginnen immer mit dem Wort *Lösungen*:

Absatzanalyse Sportbekleidung.xlsx

→ *Lösung Absatzanalyse Sportbekleidung.xlsx*

Jetzt wünsche ich Ihnen viel Spaß und viel Erfolg bei Ihrer Aufgabe, „Business Intelligence mit Excel“ kennenzulernen. Mit diesem Buch wird Ihnen das gelingen!

Ihr Autor

Ignatz Schels

- *Power BI* ist der neue Star am Reporting-Himmel: Das Produkt gehört zwar weder zu Excel noch zum Microsoft Office-Paket, wird aber in Zukunft eine bedeutende Rolle bei *Business Intelligence mit Excel* spielen. Excel arbeitet eng mit *Power BI* zusammen. Excel-Daten werden in Dateiform als Datenquellen zur Verfügung gestellt oder dynamisch in Berichte und Dashboards eingebunden. Mit dem Add-in *Power BI Publisher* werden Excel-Daten eng mit *Power BI* verknüpft.

1.2.1 Power Query

Power Query ist das Datenabfrage-Tool, das für den ETL-Prozess generiert wurde und diesen mit genialen Werkzeugen unterstützt. ETL steht für *Extract, Transform* und *Load*. Mit Power Query werden die Daten aus unterschiedlichsten Datenquellen importiert (Extract), bereinigt, umgewandelt, modelliert, neu berechnet oder kombiniert (Transfer) und in das Datenmodell für die BI-Analyse geladen (Load). Aus dem Datenmodell entstehen im letzten Schritt Berichte und Dashboards, aber dafür ist wieder ein anderes Tool (Power BI) zuständig.

Power Query-Erzeugnisse generieren dynamische Abfragen auf die Daten. Für eine einmal modellierte Datenmenge genügt ein Knopfdruck, und alle Änderungen aus den Quelldaten werden automatisch aktualisiert.

Power Query kommt mit einer eigenen Programmiersprache namens M. Alle Schritte, die im Rahmen des ETL-Prozesses nötig sind, um die Daten bereitzustellen, werden in einer Art Protokoll aufgezeichnet, und für die Digitalisierung der einzelnen Aktionen verwendet M eine spezifische Sprache mit entsprechend großem Sprachumfang. Wer sich bereits mit VBA, der Makrosprache von Excel, beschäftigt hat, kennt den Makrorecorder. Er zeichnet die Aktionen des Anwenders auf und wandelt sie in VBA-Befehle um. Das Ergebnis ist ein Makro, das alle Aktionen „auf Knopfdruck“ wieder ausführt.

Power Query-Abfragen entstehen ähnlich: Der Anwender importiert Daten, bereinigt diese, filtert und sortiert, passt Datentypen an, entfernt Spalten und berechnet zusätzliche Spalten. Alles wird Schritt für Schritt aufgezeichnet, und zum Schluss hält der Anwender eine Abfrage in den Händen, die nur noch aktualisiert werden muss, um alle diese Schritte wieder neu auszuführen. Die Programmiersprache M bietet darüber hinaus noch zusätzlich Elemente wie Variablen, Parameter, logische Strukturen und eine Art Schleifen an, mit denen die aufgezeichneten Abfragen noch verfeinert und verbessert werden können.

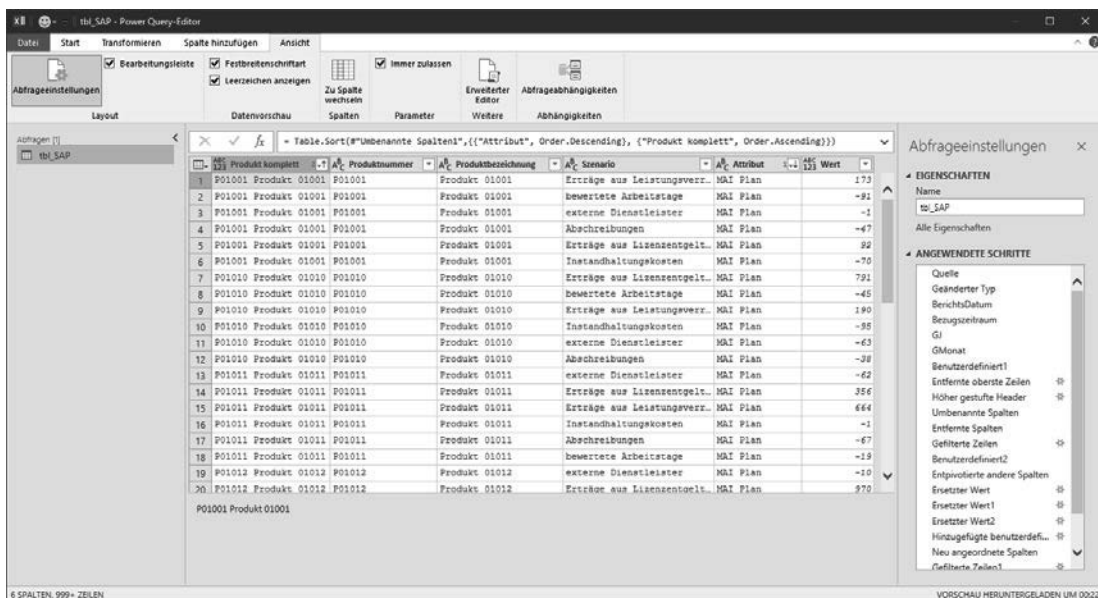


Bild 1.1 Der Power Query-Editor

1.2.2 Power Pivot

Mit Power Pivot bietet Excel ein Tool für die Integration großer Datenmengen aus unterschiedlichsten Quellen (Datenbanken, Excel-Mappen, Text/CSV, Internet u. a.). Power Pivot verwaltet die Daten in Form von Tabellen. Verknüpfungen in importierten Datenbank-Daten werden automatisch erkannt und übernommen, mit Hilfe einer Diagrammansicht können unterschiedlichste Datenquellen aber auch manuell über Schlüsselfelder verknüpft werden. Damit stellt Power Pivot eine besonders nützliche Komponente für Self Service Business Intelligence zur Verfügung.

Power Pivot ist keine Weiterentwicklung der guten alten PivotTable. Technisch gesehen ist Power Pivot die Oberfläche eines SQL-Datenbankservers, und das Datenmodell, auf dem das Tool aufbaut, ist eine OLAP-Datenbank, die unsichtbar im Hintergrund aufgebaut wird. Wer sich entscheidet, in Zukunft mit Power Pivot zu arbeiten, sollte sich deshalb gleich mit den Prinzipien des Datenbankdesigns anfreunden, denn darauf basiert das Tool. Felder, KPIs, Hierarchien, 1:n-Beziehungen, Schlüsselfelder, linker und rechter Join, Datenbank-Speech – mit diesen Dingen sollten Sie sich vertraut machen. Dazu kommt die neue Formelsprache DAX mit englischsprachigen (!) Funktionen, die oft identisch mit vertrauten Excel-Funktionen (MAX, MIN, SUM) und doch so verschieden (RELATED, SUMX, PREVIOUSPERIOD) sind. Formeln heißen Measures, PivotTables sind die einzige Schnittstelle zu Excel. Das vertraute Tabellenblatt spielt mit Power Pivot keine Rolle mehr, es gibt nur noch das Datenmodell und die PivotTables.

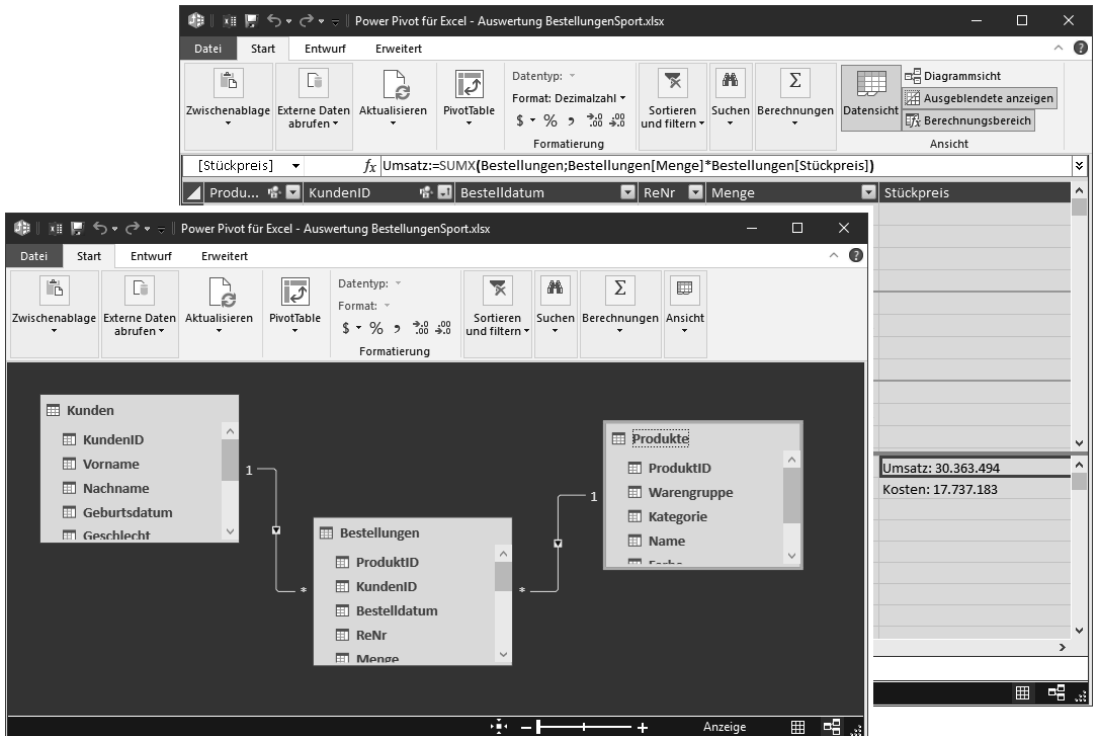


Bild 1.2 Power Pivot mit Daten- und Diagrammansicht

1.2.3 Power BI

Power BI ist im Verbund der BI-Tools für die Visualisierung der Daten zuständig. Aber damit nicht genug – das Programm bietet sogar die Funktionalität der beiden anderen Tools Power Query und Power Pivot. In der Praxis lässt sich mit Power BI also ein komplettes BI-System ganz ohne Excel-Unterstützung aufbauen. Power BI kann Daten aus unterschiedlichsten Quellen importieren, bietet die Möglichkeit, Daten ähnlich wie Power Query zu modellieren und zu transferieren, und genauso wie in Power Pivot können Tabellen auch über Schlüssel-felder relational verknüpft werden. Die Skriptsprache M steht ebenso wie in Power Query zur Verfügung, und das Datenmodell lässt sich über DAX-Funktionen mit zusätzlichen Kalkulationen erweitern.

Für die Visualisierung stehen zahlreiche Diagrammelemente, vom Balkendiagramm bis zur Geo-Map, und unterstützende Werkzeuge (Datenschnitte, Bild- und Zeichenwerkzeuge u. a.) zur Verfügung. Das Angebot an Visualisierungswerkzeugen wächst ständig, Entwickler können diese sogar selbst entwerfen und individuellen Anforderungen anpassen. Berichte können online und mobil abgefragt, bearbeitet, zu Dashboards zusammengefasst und mit Kollegen geteilt werden.

2

Grundlagen, Verfahren und Konzepte

Business Intelligence mit Excel ist mehr als Excel. Neue Verfahren, neue Werkzeuge und neue Konzepte halten Einzug in die altherwürdige Tabellenkalkulation. Wer sich noch auf dem Wissensstand von vor 10 Jahren ausruht und seinen Horizont auf Zellen, Tabellenblättern und Arbeitsmappen mit Formatierungen und einigen Funktionen begrenzt, sollte umdenken. Das Wissen um grundlegende Datenbankkonzepte und -formate wird für Business Intelligence ebenso gebraucht wie der sichere Umgang mit den BI-Tools.

■ 2.1 Relationale und multidimensionale Datenbanken

Die relationale Datenbank ist eine Erfindung aus den 70er-Jahren. Dr. Edgar F. Codd, ein Programmierer bei IBM, erfand das Modell der relationalen Datenbank. Er verwendete die Abfragesprache SEQUEL, aus der SQL hervorging, die Sprache, die heute noch Kern und Basis aller Datenbanksysteme ist. Edgar F. Codd entwickelte auch die Normalformen und die 12 Evaluierungsregeln als Anforderung für ein OLAP-System (Online Analytic Processing). Oracle kam als erstes Unternehmen im Jahr 1979 mit einem relationalen Datenbankmanagementsystem auf den Markt.

Mitte der 90er-Jahre nutzte jedes größere Unternehmen relationale Datenbanken. Aber mit der Zeit tauchte ein Problem auf: Die Datenbanken mit relationalen Verknüpfungen waren zwar ideal geeignet, um Daten auch in größeren Mengen zu speichern. Für die Erstellung von Berichten aus Transaktionsdaten erwiesen sich die Datenbanken jedoch als zu schwerfällig. Musste zum Beispiel ein Bericht über monatliche Umsätze erstellt werden, suchte das System in Millionen von Transaktionen, um eine einzelne Summe zu ermitteln. Nicht selten ging dabei die gesamte Serverwelt „in die Knie“, und an schnelle, Just-in-time-Informationen war gar nicht zu denken.

4

Abrufen und Transformieren (Power Query)

Unter dem Codenamen Data Explorer tauchte 2013 die erste Version von Power Query auf. Ebenfalls in den SQL-Laboratories entwickelt, wird das Tool wieder für Excel 2010 als Add-in zum Download angeboten, ab 2013 gehört es zum Programm. In Excel 2016 verschwand der Name Power Query, das Daten-Register enthält eine Gruppe *Abrufen und Transformieren* mit den Werkzeugen von Power Query.

Power Query holt die Daten aus unterschiedlichen Datenquellen, bereitet sie für die Analyse auf und speichert sie entweder im internen Datenmodell oder als „intelligente“ Excel-Tabellen. Aus diesen werden sie über PivotTables oder mit PowerMap visualisiert bzw. nach Power BI transferiert.

Die Auswahl der Datenquellen reicht von der einfachen Textdatei bis zum Cube, von SQL Server bis Azure Marketplace. Selbst die Einbindung des Active Directory und des Exchange-Mailservers ist kein Problem, Power Query kann damit auch Dokumentstrukturen und Mails, Kontakte, Termine etc. analysieren. Mit der Möglichkeit, Facebook-Daten abzurufen, stößt Power Query in den Bereich Social Intelligence vor.

Mit dem integrierten Editor analysiert Power Query die Daten, formatiert sie, löscht nicht Benötigtes und kombiniert bei Bedarf mehrere Abfragen über relationale Beziehungen. Alle Schritte werden mit einer Art Makrorecorder aufgezeichnet und können in diesem editiert werden.

■ 4.1 Menüband und Symbole

Warum Microsoft den einprägsamen Begriff Power Query abgeschafft hat, ist unverständlich, aber nicht zu ändern. Für Excel mit der Versionsnummer 2010 und 2013 gab es Power Query bereits als Add-in, es kann immer noch kostenlos bei Microsoft im Web heruntergeladen werden. Ab Excel 2016 ist Power Query Bestandteil der Menüoberfläche, der Begriff ist verschwunden, die Menübandgruppe, in der alle Symbole von Power Query bereitstehen, heißt *Abrufen und Transformieren*.

In Excel 2016 wurde das Power Query zunächst in der Gruppe *Abrufen und Transponieren* untergebracht, die alten MS Query-Befehle standen weiterhin in der Gruppe *Externe Daten*

abrufen zur Verfügung. Mit dem ersten Update verschwand dann die Gruppe *Externe Daten abrufen*, die MS Query-Befehle wurden in Power Query integriert.

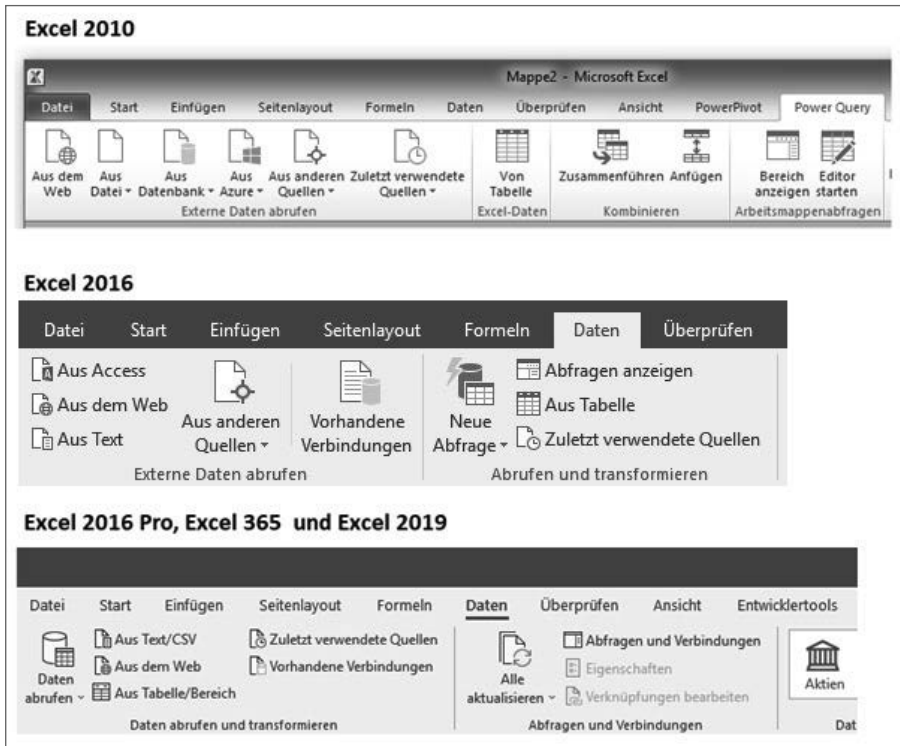


Bild 4.1 Von Power Query zu „Daten abrufen und transformieren“

■ 4.2 ETL – der Prozess

Power Query ist das BI-Werkzeug in Excel, mit dem Daten aus externen Quellen in die Analyse eingebunden werden. Hinter dieser pauschalen Aussage verbirgt sich ein umfassender, aufwendiger Prozess, den es in dieser Form im Umfeld von Excel oder Office noch nicht gab. Bisher war das Analysieren von Daten eine Abfolge manueller Tätigkeiten:

Daten manuell importieren: Excel-Arbeitsmappen oder CSV-Dateien werden geöffnet, Tabellenblätter angesteuert, Daten kopiert oder verschoben. Textdateien müssen über den Text-Assistenten importiert werden, Excel schaltet diesen automatisch ein beim Versuch, eine Textdatei zu öffnen. Dann muss das Format und das Trennzeichen bestimmt werden, im schlechteren Fall sind die Spalten manuell zu trennen. Nach Auswahl der Zahlenformate für die einzelnen Spalten wird der Text importiert, kann in Spaltenform analysiert und im Excel-Format gespeichert werden.

Daten verknüpfen: Datenbankdaten aus Access, SQL Server oder anderen Quellen werden per MS Query und ODBC verknüpft, Text/CSV und Excel-Inhalte können ebenfalls dynamisch eingebunden werden. Für die Analyse von Internet/Intranet-Daten gibt es einen Web-Browser, der HTML-Elemente auf Webseiten identifiziert und die Möglichkeit bietet, für diese eine Verknüpfung einzurichten.

Daten aufbereiten: Sind die Daten manuell oder per ODBC bereitgestellt, beginnt für den Anwender die eigentliche Arbeit: Überflüssige Zeilen und Spalten löschen, Daten konvertieren, Datenmüll (Sonderzeichen, überflüssige Leerzeichen etc.) beseitigen, Spalten durchgehend füllen usw. Und wer schon mal versucht hat, Daten aus unterschiedlichen Quellen zu einem Datenbestand als Basis für eine PivotTable/PivotChart-Auswertung zusammenzuführen, wird sicher schon an seine Grenzen gestoßen sein.

ETL ist die Abkürzung für *Extract, Transform, Load* (*Extraktion, Transformation, Laden*) und bezeichnet den Prozess, bei dem Daten aus mehreren gegebenenfalls unterschiedlich strukturierten Datenquellen in einer Zieldatenbank vereinigt werden (vergleiche Wikipedia, <https://de.wikipedia.org/wiki/ETL-Prozess>). Was bisher nur für das Beladen von Data Warehouses eine Rolle spielte, haben die Power Query-Entwickler aufgenommen und in die Tat umgesetzt:

Extraktion: Daten werden aus der Datenquelle extrahiert und für die Transformation bereitgestellt. Das Ganze muss synchron passieren, d. h. jede Änderung an der Datenquelle wird automatisch weitergegeben.

Transformation: Die Daten aus unterschiedlichen Quellen werden in ein einheitliches Datenschema transferiert und gleichzeitig bereinigt. Zur Transaktion gehört die Anpassung der Datentypen (z. B. Datumsformate), die Eliminierung von Duplikaten, die Umrechnung von Maßeinheiten und die Anreicherung der Daten mit Zusatzinformationen (z. B. demografische Daten aus dem Web).

Laden: Die Daten werden aus dem Arbeitsbereich in den Zielbereich (Data Warehouse) eingebracht, gleichzeitig wird eine Versionshistorie mit Protokoll der Änderungen angefertigt.

Power Query hat das ETL-Prinzip ziemlich gut umgesetzt, wobei der Schwerpunkt auf der Transformation liegt: Was der Anwender bisher manuell umsetzte, erledigt Power Query in einer Abfrage. Alle Aktionen müssen nur einmal ausgeführt werden, Power Query speichert diese Schritte und führt sie mit jeder Aktualisierung der Abfrage wieder aus. Die Abfrage wird protokolliert, das Protokoll ist eine Art Ablaufcode, der vom Anwender eingesehen, abgeändert und sogar mit einer eigenen Programmiersprache (M) verfeinert und erweitert werden kann. Ein geniales Konzept, mit dem das Konzept der Datenanalyse mit Excel neu geschrieben wird.

Power Query ist deshalb auch Bestandteil und Basis von Power BI, dem BI-Tool von Microsoft, das umfangreiche Reports und Visualisierungen ermöglicht (siehe Kapitel 8).

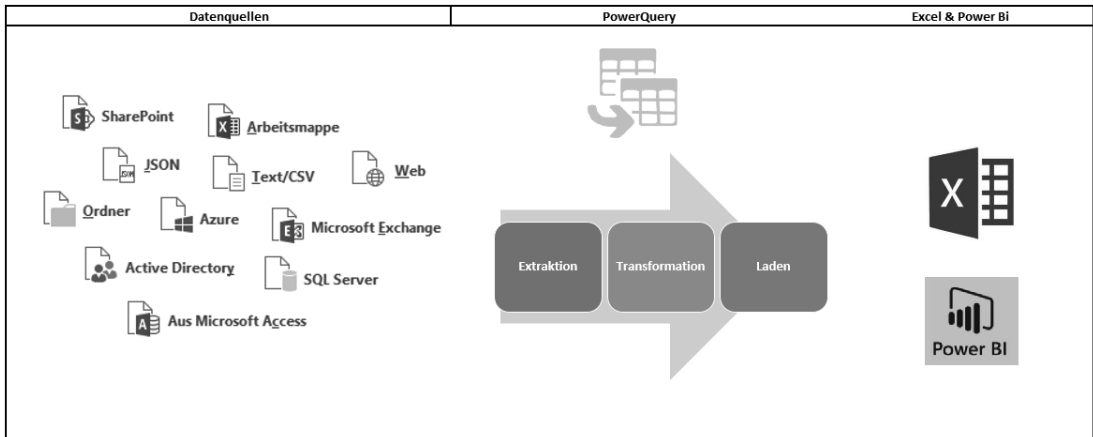


Bild 4.2 Der ETL-Prozess

■ 4.3 Power Query installieren

Power Query läuft noch unter Windows 8, aber versuchen Sie besser nicht, das Tool mit diesem (oder einem noch älteren) Alt-System zu betreiben. Windows 7 oder Windows 2008 Server sind Mindestanforderung. Microsoft .NET 3.5 mit Service Pack 1 muss installiert sein, der Internet-Browser sollte Internet Explorer heißen und die Versionsnummer 9 oder höher tragen (in Windows 10 natürlich Edge, der Nachfolger).

In der Version 2010/2013 ist Power Query standardmäßig nicht enthalten, das Tool lässt sich aber einfach integrieren. Es steht als (kostenloser) Download bei Microsoft zur Verfügung und ist in wenigen Minuten installiert:

1. Suchen Sie mit Google nach „Power Query Excel 2010“. Öffnen Sie die Seite, die mit dem Link auf www.microsoft.com/de-de/Downloads angeboten wird.
2. Achten Sie darauf, dass die passende Sprachversion (Deutschland) eingestellt ist und suchen Sie die Version, die zu Ihrer Office-Version passt (32/64 Bit). Wenn Sie nicht wissen, ob Sie eine 32-Bit- oder 64-Bit-Version verwenden, sehen Sie unter *Datei/Konto/Info zu Excel* nach.
3. Starten Sie die Installation über die heruntergeladene MSI-Datei und bestätigen Sie alle Schritte des Assistenten. Nach Abschluss der Installation finden Sie einen neuen Reiter Power Query im Menüband.
4. Sollte die neue Registerkarte nicht zu sehen sein, wählen Sie *Datei/Optionen/Add-Ins*. Suchen Sie unter *Verwalten* die *COM-Add-Ins* und kreuzen Sie, nachdem Sie auf die Schaltfläche *Gehe zu* geklickt haben, den Eintrag *Power Query* an.



HINWEIS: Power Query unterscheidet sich in Excel 2010 in der Oberfläche vom Nachfolger, und der Abfrage-Editor enthält nicht alles, was die Version für Excel 2016/2019 kann.

■ 4.4 Daten abrufen und transformieren

In der ersten Version Excel 2016 gab es noch im Menüband eine Gruppe *Externe Daten abrufen*, in der Excel alle Datenquellen für das ältere Abfragesystem MS Query zur Auswahl stellte. Excel war als Datenquelle nicht dabei, unter *Aus anderen Quellen* konnte MS Query abgerufen werden, und erst der nachfolgende Dialog stellte den ODBC-Treiber für Excel zur Auswahl. Diese Gruppe wurde mit dem Januar-2017-Update von Office 365 herausgenommen. In der Standard-Version von Excel 2016 ohne automatische Updates wird sie aber nach wie vor angeboten sein.

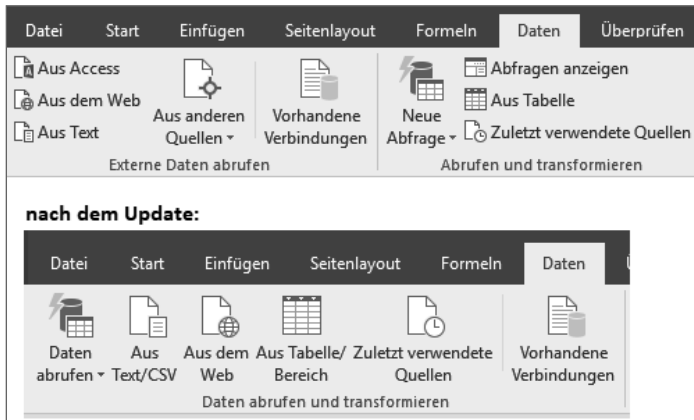


Bild 4.3 Unterschiedliche Menüband-Einträge in den Excel-Versionen

Excel 2016/2019 bietet in der Gruppe *Daten abrufen und transformieren* ein Listensymbol für alle Datenquellen und je ein Symbol für *Text/CSV*, *Web*, *Tabelle/Bereich* und *Zuletzt verwendete Quellen*. Klicken Sie auf den Pfeil des Symbols *Daten abrufen*, um alle Datenquellen aufzulisten:

4.4.1 Aus Datei

Wählen Sie *Aus Arbeitsmappe*, um eine Excel-Mappe als Datenquelle zu aktivieren. Der Dateidialog bietet alle Mappen mit den Dateiendungen *.xlsx*, *.xlsm*, *.xltx* und *.xlsb* an, auch die älteren Dateiformate *XLS* und *XLT* sowie die Add-ins (*XLAM*, *XLA*) stehen zur Auswahl. Der Abfrage-Assistent öffnet die Datei und bietet alle darin enthaltenen Tabellenblätter zur Auswahl an.

Aus Text/CSV bietet den Dateiauswahldialog für Text- und CSV-Dateien an. Suchen Sie mit **.**, falls Ihre Textdatei eine andere Dateiendung hat als *.txt*, *.csv* oder *.prn*. XML-Dateien werden ebenso angeboten wie JSON (JavaScript-Dateien).

Entscheiden Sie sich für *Aus Ordner*, können Sie einen Ordner angeben oder suchen, aus dem alle Dateien in die Abfrage mit einbezogen werden. Voraussetzung ist natürlich, dass alle Dateien in diesem Ordner die gleiche Struktur haben. Mit *Aus SharePoint-Ordner* er-

Power Pivot – Daten modellieren im Datenmodell

■ 5.1 Grundlagen

Es begann mit einem internen Whitepaper, verfasst 2006 von Microsofts SQL Server-Analysten Amir Netz. Die „BI-Sandbox“ sollte BI-Anwendungen einfacher machen, relationale und multidimensionale Datenbanken als Quellen integrieren und ein Reporting-Tool beinhalten. Die Microsoft-Datenbank Access war als Plattform anvisiert.

Microsoft baute ein Team für die Umsetzung der Idee auf, nannte das Projekt Gemini und das Resultat hieß Power Pivot und war Teil des 2010 ausgelieferten SQL Server-Release 2008 R2. Parallel dazu wurde eine Version für Excel 2010 und eine SharePoint-Version gebaut. Excel 2010 hatte vorher noch ein besonders nützliches Werkzeug aus der Power Pivot-Entwicklung mitbekommen, den Datenschnitt (Slicer).

Darauf hatten Excel-Anwender lange gewartet, denn der „alte“ Berichtsfilter der PivotTables ist nicht der Weisheit letzter Schluss, er kann zwar mehrere Elemente filtern, zeigt aber die Filterkriterien nicht mehr an (das erklärt übrigens auch, warum Excel 2007 keine Datenschnitte hat – Power Pivot war 2007 einfach noch nicht fertig).

5.1.1 Das kann Power Pivot

Power Pivot ist die Lösung, wenn Datenmengen ins Spiel kommen, die Excel nicht bewältigen kann. Die Größe des Tabellenblattes täuscht: 1 048 576 Zeilen und 16 384 Spalten stehen zwar zur Verfügung, aber ab einigen 100 000 Datensätzen (je nach Speichergröße) wird es bei Excel eng mit der Performance, und PivotTables verweigern ihre Dienste bei großen Datenmengen noch viel früher.

- Power Pivot kann umfangreiche Datenmengen importieren, mehrere Millionen Datensätze in akzeptabler Geschwindigkeit aus ERP-Systemen, unterschiedlichsten Datenquellen (Datenbanken, OLAP-Cubes, CSV, Text etc.) laden und auswerten.
- Power Pivot ermöglicht Auswertungen mit den „Bordmitteln“ von Excel: PivotTable-Berichte, PivotCharts, Tabellen, Formeln und Funktionen. Für Berechnungen steht die Formelsprache DAX zur Verfügung.

- Power Pivot erkennt Verknüpfungen in relationalen Datenbanken, Measures und Dimensionen in Cubes. Beziehungen lassen sich verwalten, Beziehungen können in Power Pivot erstellt werden, auch zwischen Daten aus unterschiedlichen Quellen.
- Power Pivot-Analysen lassen sich mit wenigen Klicks auf SharePoint-Server-Seiten veröffentlichen. Ressourcennutzung lässt sich nachverfolgen, für die Sicherheit sind umfangreiche Features verfügbar.

5.1.2 Technische Voraussetzungen

Power Pivot läuft auf Rechnern mit 32-Bit- und 64-Bit-Architektur, der Prozessor muss mindestens mit 500 MHz takten. Die Größe des Arbeitsspeichers ist natürlich maßgebend für die Geschwindigkeit, für Power Pivot sind 2 GB RAM erforderlich, bis zu 4 GB werden unterstützt. Das Add-in belegt 25 MB, die erste Power Pivot-Tabelle belegt 33 MB.

Das Betriebssystem kann Windows XP SP3 (32 Bit) oder Vista SP2 sein, Windows Server 2008 SP2 oder Windows 7 ist empfohlen. Das NET Framework 3.5 SP1 muss installiert werden.

5.1.3 Office-Version

Power Pivot läuft ab Excel 2010, bei der Installation von Office 2010 müssen alle gemeinsam genutzten Office-Tools und die .NET-Programmunterstützung installiert werden. In der Systemsteuerung können Sie das Setup für Office 2010 aktivieren und fehlende Tools nachinstallieren. Mit Excel 2007 können Arbeitsmappen mit Power Pivot-PivotTables geöffnet und verändert werden, ein Ändern der Datenquelle ist nicht möglich.

Nicht enthalten ist Power Pivot in diesen Office-Versionen:

- Office 365 Education, Business Essentials und Enterprise E1
- Office 2013 Home & Student, Home & Business
- Office-Lizenzversionen (Einmalkauf) Home & Student 2016 oder Home & Business 2016
- Office für Mac/Android

Für Power Pivot brauchen Sie also eine dieser Office-Versionen:

- Office 365 Pro Plus, Enterprise E3 oder Enterprise E5
- Office 2013 Professional Plus
- Office Professional 2016/2019 oder Professional Plus
- Alle eigenständigen Versionen von Excel 2013/2016/2019

Das Add-in, das für Excel 2010 zum kostenlosen Download zur Verfügung steht, funktioniert nicht mit Office 2013/2016/2019.

	A	B	C	D	E	F	G	H
1	Bestelldatum	Umsatz						
2	01.01.2018	13.520						
3	02.01.2018	18.629						
4	03.01.2018	13.497						
5	04.01.2018	4.363						
6	05.01.2018	14.623						
7	06.01.2018	15.733						
8	07.01.2018	18.142						
9	08.01.2018	15.558						
10	09.01.2018	13.977						
11	10.01.2018	13.255						
12	11.01.2018	20.862						

Bild 7.24 Eine Zeitachse für das Datumsfeld

■ 7.7 Berechnete Felder

In der Praxis enthalten die Datenquellen für PivotTables selten alle Felder, die benötigt werden. Zusätzliche Berechnungen wie Boni für Kunden oder Mitarbeiter, Aufschläge und Zuschläge bei Aufträgen und Bestellungen oder Abzüge bei Rechnungen lassen sich aus den vorliegenden Daten errechnen oder durch Kombination mit zusätzlichen Daten ermitteln (Beispiel: Zuschlagstabellen).

Entscheidend für die Auswertung der Daten ist die Frage, wann und wo die Berechnung zusätzlicher Felder stattfindet. Hier gibt es mehrere Möglichkeiten:

7.7.1 Berechnete Felder und Elemente in der PivotTable

Das ist mittlerweile die schlechteste aller Lösungen. Berechnungen lassen sich zwar als zusätzliche Felder im PivotTable-Layout integrieren, die Auswahl an Kontrollstrukturen ist aber beschränkt, Funktionen, Vergleiche oder Verweise lassen sich gar nicht nutzen. Basiert die PivotTable auf dem Datenmodell, lässt die PivotTable keine berechneten Felder und Elemente mehr zu, hier muss auf M-Funktionen oder DAX-Funktionen zurückgegriffen werden.

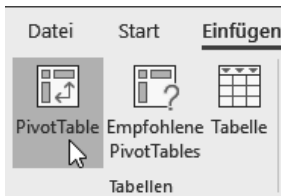
Berechnete Felder können Feldnamen, mathematische Operatoren und Funktionen aus dem Excel-Angebot enthalten. Das Angebot ist aber sehr beschränkt, erlaubt sind nur Funktionen, die keine wechselnden Argumente haben. Auch Matrixfunktionen, Zellbezüge oder Bereichsnamen funktionieren nicht. Beschränken Sie sich deshalb in berechneten Feldern auf reine Arithmetik.

Im Beispiel liegt eine Liste mit Verkäuferumsätzen im ersten Quartal vor. Weisen Sie den Umsätzen in der PivotTable über ein berechnetes Feld jeweils 25 % Provision zu und berechnen Sie mit berechneten Elementen die Quartalssummen.

Aktivieren Sie *Umsatz Verkäufer.xlsx*. Die Liste enthält die Namen, die Monate des ersten Quartals und die Umsatzzahlen.

	A	B	C
1	Name	Monat	Umsatz
2	Hugo Tüchtig	Jan	2.300
3	Hugo Tüchtig	Feb	3.200
4	Hugo Tüchtig	Mrz	4.100
5	Ernst Fleißig	Jan	5.000
6	Ernst Fleißig	Feb	5.500
7	Ernst Fleißig	Mrz	6.500
8	Berta Emsig	Jan	2.000
9	Berta Emsig	Feb	3.000
10	Berta Emsig	Mrz	4.000

Wählen Sie *Einfügen/Tabellen/PivotTable*.

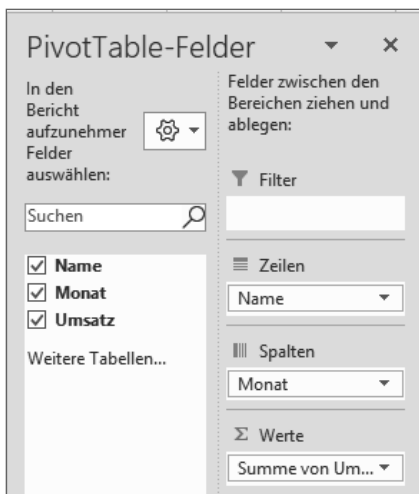


Legen Sie eine PivotTable mit der Liste als Basis an und ziehen Sie die Felder im Layout in die passenden Bereiche:

Zeilenbereich: Name

Spaltenbereich: Monate

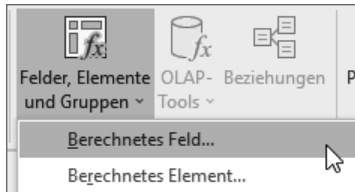
Wertebereich: Umsatz



Das Ergebnis: Die PivotTable präsentiert die Umsatzsummen für Verkäufer und Monat.

H	I	J	K	L
Summe von Umsatz	Spaltenbeschriftungen			
Zeilenbeschriftungen	Jan	Feb	Mrz	Gesamtergebnis
Berta Emsig	2000	3000	4000	9000
Ernst Fleißig	5000	5500	6500	17000
Hugo Tüchtig	2300	3200	4100	9600
Gesamtergebnis	9300	11700	14600	35600

Wählen Sie *PivotTable-Tool/Analysieren/Berechnungen/Felder, Elemente und Gruppen/Berechnetes Feld*.

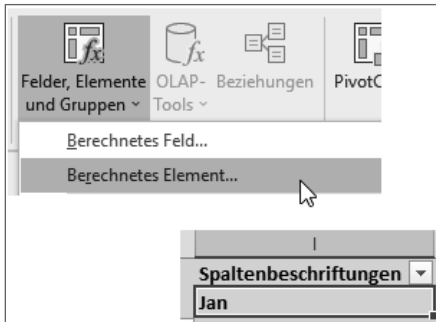


Legen Sie ein berechnetes Feld an. Das Umsatzfeld können Sie per Doppelklick in die Formel holen:

Name: Provision
Formel: =Umsatz*1,25

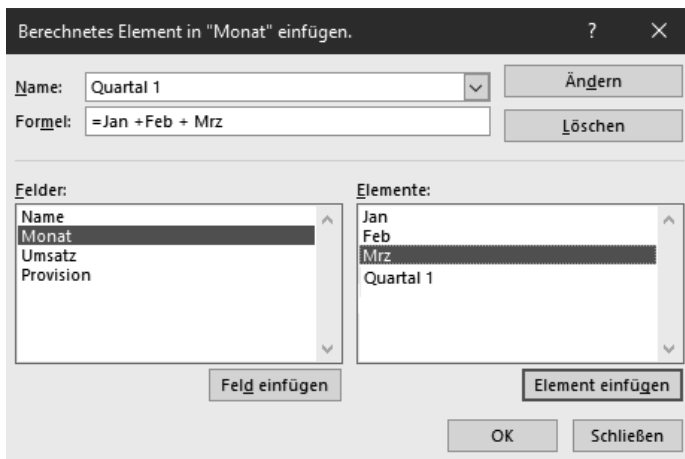


Setzen Sie den Zellzeiger in einen der Monatsnamen im Spaltenbereich und wählen Sie *PivotTable-Tool/Analysieren/Berechnungen/Felder, Elemente und Gruppen/Berechnetes Element*.



Erstellen Sie ein berechnetes Element. Die beteiligten Feldelemente (hier die Feldnamen des Feldes Monat) holen Sie per Doppelklick in die Formel.

Name: Quartal 1
Formel: = Jan+Feb+Mar



Das Ergebnis ist eine PivotTable mit einem zusätzlichen Feld für die Provision und einem neuen Element für das Feld *Monat*. Ziehen Sie die Felder *Name* und *Monat* in den Zeilenbereich, sehen Sie dieses Ergebnis. Die Aggregatfunktion für berechnete Felder ist immer SUMME.

Zeilenbeschriftungen	Summe von Umsatz	Summe von Provision
Berta Emsig	18.000	22.500
Jan	2.000	2.500
Feb	3.000	3.750
Mrz	4.000	5.000
Quartal 1	9.000	11.250
Ernst Fleißig	34.000	42.500
Jan	5.000	6.250
Feb	5.500	6.875
Mrz	6.500	8.125
Quartal 1	17.000	21.250
Hugo Tüchtig	19.200	24.000
Jan	2.300	2.875
Feb	3.200	4.000
Mrz	4.100	5.125
Quartal 1	9.600	12.000
Gesamtergebnis	71.200	89.000

7.7.2 Berechnete Felder mit strukturierten Verweisen

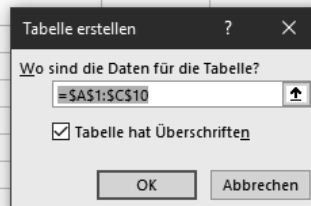
Da berechnete Felder und Elemente sehr begrenzt in ihren Fähigkeiten sind, sollten Sie auf andere, neuere Techniken zurückgreifen. Wer sich gut auskennt mit Excel-Funktionen und wenig Zeit oder Lust hat, DAX oder M zu lernen, kann Berechnungen auch in der Tabelle durchführen. Die Daten holen Sie sich dynamisch mit Power Query, zusätzliche Felder und Elemente werden einfach angefügt oder zwischen die Abfragespalten eingefügt. Die Formeln passen sich natürlich an, wenn die Abfrage aktualisiert wird.

Verwenden Sie keine Listen und keine Bezüge (\$A\$1), denn damit erhalten Sie keine dynamischen Auswertungen. In Tabellen kommen nur strukturierte Verweise zum Einsatz, und die Formeln gelten automatisch für die gesamte Spalte und werden in dieser auch dynamisch verwaltet.

Hier unser Beispiel mit den Umsatzzahlen mit zusätzlichen Berechnungen über strukturierte Verweise:

Holen Sie die Liste am besten per Power Query aus der Datei *Umsatz Verkäufer.xlsx* oder erstellen Sie mit **Strg+T** manuell eine Tabelle.

	A	B	C	D	E	F
1	Name	Monat	Umsatz			
2	Hugo Tüchtig	Jan	2.300			
3	Hugo Tüchtig	Feb	3.200			
4	Hugo Tüchtig	Mrz	4.100			
5	Ernst Fleißig	Jan	5.000			
6	Ernst Fleißig	Feb	5.500			
7	Ernst Fleißig	Mrz	6.500			
8	Berta Emsig	Jan	2.000			
9	Berta Emsig	Feb	3.000			
10	Berta Emsig	Mrz	4.000			
11						
12						
13						



Legen Sie eine neue Spalte *Provision* an. Achten Sie darauf, dass die Spalte direkt neben der letzten Spalte steht, damit sie in die Tabelle integriert wird.

	A	B	C	D
1	Name	Monat	Umsatz	Provision
2	Hugo Tüchtig	Jan	2.300	
3	Hugo Tüchtig	Feb	3.200	
4	Hugo Tüchtig	Mrz	4.100	
5	Ernst Fleißig	Jan	5.000	
6	Ernst Fleißig	Feb	5.500	
7	Ernst Fleißig	Mrz	6.500	
8	Berta Emsig	Jan	2.000	
9	Berta Emsig	Feb	3.000	
10	Berta Emsig	Mrz	4.000	

Schreiben Sie die Formel, die eine Provision von 25 % (zuzüglich) berechnet. Für den strukturierten Verweis klicken Sie einfach auf die erste Zelle in der Spalte *Umsatz*.

=[@Umsatz]*1,25

✕		✓	<i>f_x</i>	=[@Umsatz]*1,25	
C		D		E	
Umsatz		Provision			
2.300		=[@Umsatz]*1,25			

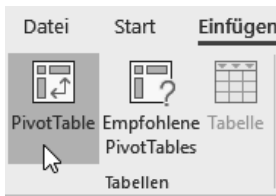
Die Formel wird automatisch über die gesamte Länge der Tabelle kopiert. Sie wird auch selbstständig kopiert, wenn neue Zeilen hinzukommen und bei entfernten Zeilen gelöscht.

✖		✔		fx		=[@Umsatz]*1,25	
C		D		E			
Umsatz		Provision					
2.300		2.875					
3.200		4.000					
4.100		5.125					
5.000		6.250					
5.500		6.875					
6.500		8.125					
2.000		2.500					
3.000		3.750					
4.000		5.000					

Für weitere Berechnungen verwenden Sie Funktionen von Excel. Hier wird das Quartal etwas aufwendig berechnet, weil der Monat nur in Textform zur Verfügung steht.

=AUFRUNDEN(MONAT(DATWERT("1."&[@Monat]&" "&JAH(HEUTE())))/3;0)					
D	E	F	G	H	I
ion	Quartal				
2.875	1				
4.000	1				
5.125	1				

Legen Sie mit *Einfügen/Tabelle/PivotTable* eine PivotTable mit der Tabelle als Basis an.

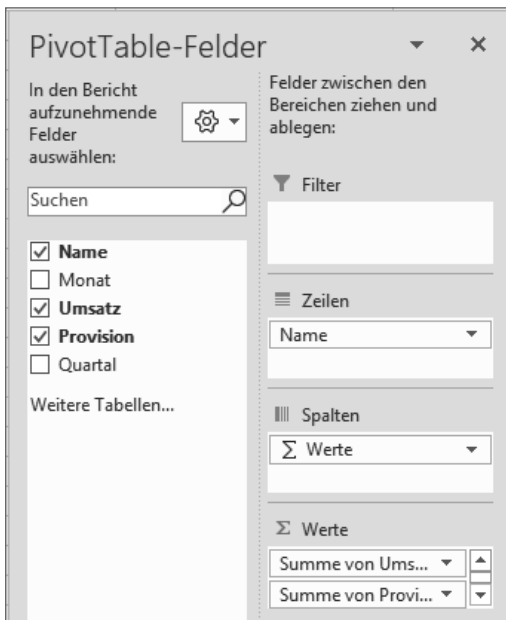


Ziehen Sie in der Feldliste die Felder an die passende Position:

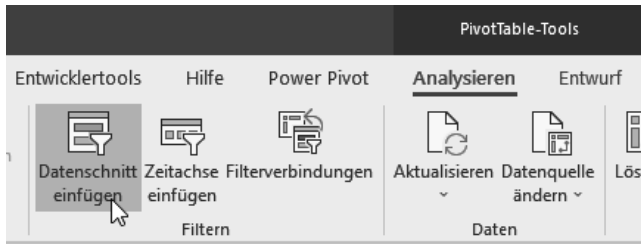
Zeilenbereich: Name

Wertebereich: Summe von Umsatz und Summe von Provision

Die Quartalsumme lässt sich hier nicht gleichzeitig auswerten.



Fügen Sie mit *PivotTable-Tools/Analysieren/Filtern/Datenschnitt einfügen* einen Datenschnitt ein. Kreuzen Sie das Feld *Quartal* an.



Jetzt lässt sich die PivotTable über den Datenschnitt nach Quartalen filtern.

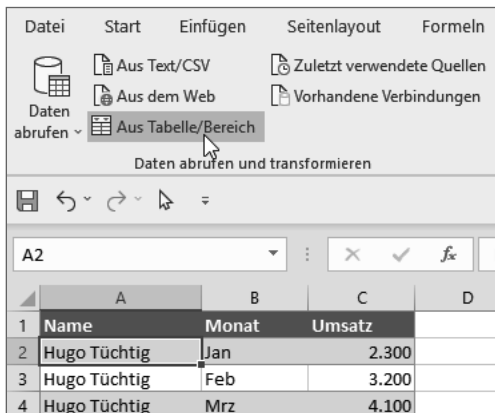
K	L	M
Quartal		
1		
Name	Summe von Umsatz	Summe von Provision
Berta Emsig	9000	11250
Ernst Fleißig	17000	21250
Hugo Tüchtig	9600	12000
Gesamtergebnis	35600	44500

7.7.3 Berechnete Felder mit Power Query

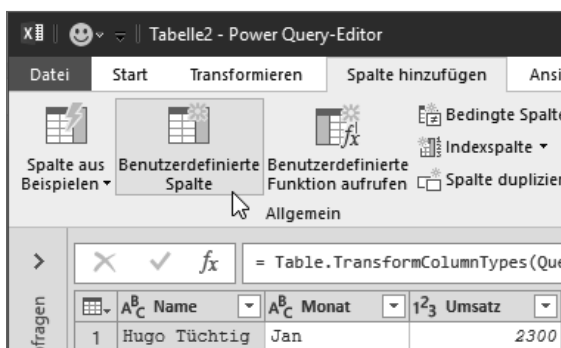
Die bessere Alternative zu berechneten Feldern in der PivotTable und zu strukturierten Verweisen ist das Anlegen benutzerdefinierter Spalten in Power Query. Die Berechnung erfolgt über die Formelsprache M, hier stehen ebenfalls alle benötigten Werkzeuge einer Programmiersprache zur Auswahl.

Unsere Beispielliste holen Sie am besten per Power Query aus der Datei *Umsatz Verkäufer.xlsx*. Alternativ dazu können Sie die Liste auch in eine Tabelle umwandeln und mit Power Query bearbeiten.

Starten Sie mit *Daten abrufen und transformieren/Aus Tabelle/Bereich* eine Power Query-Abfrage auf die Tabelle.



Legen Sie eine benutzerdefinierte Spalte an.



Geben Sie den Namen und die Formel ein.

Name: Provision
Formel: =[Umsatz]*1,25

Benutzerdefinierte Spalte

Neuer Spaltenname

Provision

Benutzerdefinierte Spaltenformel:

= [Umsatz]*1.25

Die beteiligte Spalte können Sie per Doppelklick auf den Feldnamen aus der Liste holen.

Verfügbare Spalten:

Name

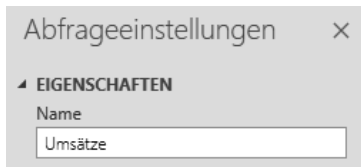
Monat

Umsatz

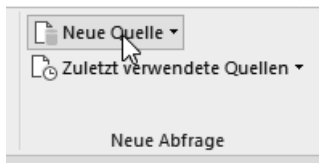
Die Provisionen werden in einer neuen Spalte berechnet.

= Table.AddColumn(#"Geänderter Typ", "Provision", each [Umsatz]*1.25)				
	A ^B C Name	A ^B C Monat	1 ² 3 Umsatz	ABC 123 Provision
1	Hugo Tüchtig	Jan	2300	2875
2	Hugo Tüchtig	Feb	3200	4000
3	Hugo Tüchtig	Mrz	4100	5125
4	Ernst Fleiß...	Jan	5000	6250
5	Ernst Fleiß...	Feb	5500	6875
6	Ernst Fleiß...	Mrz	6500	8125
7	Berta Emsig	Jan	2000	2500
8	Berta Emsig	Feb	3000	3750
9	Berta Emsig	Mrz	4000	5000

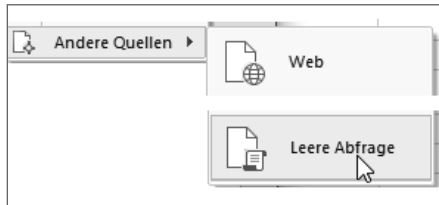
Benennen Sie die Abfrage um, geben Sie ihr den Namen *Umsätze*.



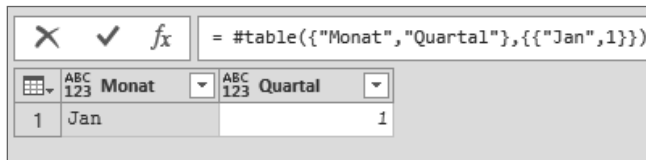
Fügen Sie über *Neue Abfrage/Neue Quelle* eine neue Abfrage ein.



Wählen Sie *Andere Quellen/Leere Abfrage*.



Schreiben Sie diese Formel, um eine neue Tabelle zu generieren:



```
= #table(
  {"Monat", "Quartal"},
  {
    {"Jan", 1}
  })
```

So sieht die gesamte Tabelle aus, komplett mit M erzeugt:

X ✓ f _x		= #table({"Monat",	
ABC 123	Monat	ABC 123	Quartal
1	Jan		1
2	Feb		1
3	Mrz		1
4	Apr		2
5	Mai		2
6	Jun		2
7	Jul		3
8	Aug		3
9	Sep		3
10	Okt		4
11	Nov		4
12	Dez		4

```

= #table({"Monat","Quartal"},
  {
    {"Jan",1}, {"Feb",1}, {"Mrz",1},
    {"Apr",2}, {"Mai",2}, {"Jun",2},
    {"Jul",3}, {"Aug",3}, {"Sep",3},
    {"Okt",4}, {"Nov",4}, {"Dez",4}
  })

```

Ändern Sie die Datentypen der beiden Spalten:

ABC 123	Monat	1.2	Quartal
1	Jan		1
2	Feb		1
3	Mrz		1
4	Apr		2

Monat=text

Quartal=number

Geben Sie der neuen Abfrage einen Namen, nennen Sie sie *MonateQuartale*.

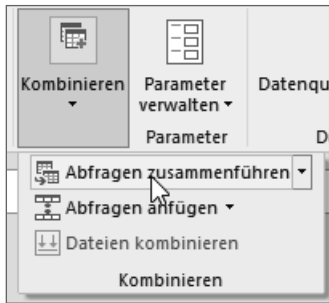
Abfrageeinstellungen X

EIGENSCHAFTEN

Name

MonateQuartale

Mit *Kombinieren/Abfragen zusammenführen* erstellen Sie eine Beziehung zwischen den beiden Abfragen.



Markieren Sie in beiden Abfragen das Feld *Umsatz* und bestätigen Sie die Beziehungsart *Inner Join*.

Zusammenführen

Wählen Sie eine Tabelle und übereinstimmende Spalten

Umsätze

Name	Monat	Umsatz	Provision
Hugo Tüchtig	Jan	2300	2875
Hugo Tüchtig	Feb	3200	4000
Hugo Tüchtig	März	4100	5125
Ernst Fleißig	Jan	5000	6250
Ernst Fleißig	Feb	5500	6875

MonateQuartale

Monat	Quartal
Jan	1
Feb	1

Ein Klick auf das Doppelpfeilsymbol erweitert die Spalte. Wählen Sie nur das Feld *Quartal*.

Umsatz ABC 123 Provision MonateQuartale

Zu erweiternde Spalten suchen

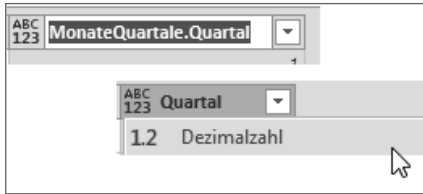
☒ Erweitern ☐ Aggregieren

☒ (Alle Spalten auswählen)

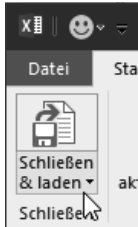
☐ Monat

☒ Quartal

Geben Sie der neuen Spalte noch den Namen *Quartal* und den Datentyp *Dezimalzahl*.



Mit *Schließen & Laden* holen Sie die beiden Abfragen in die Mappe zurück.



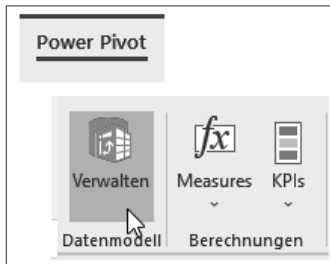
Das Ergebnis der Power Query-Abfrage wird präsentiert. Erstellen Sie eine PivotTable mit den Namen im Zeilenbereich und der Summe der Umsätze und Provisionen im Wertebereich. Das Quartal können Sie wieder als Datenschnitt zum Filtern bereitstellen.

	A	B	C	D	E	F	G	H	I
1	Name	Monat	Umsatz	Provision	Quartal		Quartal		
2	Hugo Tüchtig	Jan	2300	2875	1		1		
3	Ernst Fleißig	Jan	5000	6250	1				
4	Berta Emsig	Jan	2000	2500	1				
5	Hugo Tüchtig	Feb	3200	4000	1		Zeilenbeschriftungen	Summe von Umsatz	Summe von Provision
6	Ernst Fleißig	Feb	5500	6875	1		Berta Emsig	9000	11250
7	Berta Emsig	Feb	3000	3750	1		Ernst Fleißig	17000	21250
8	Hugo Tüchtig	Mrz	4100	5125	1		Hugo Tüchtig	9600	12000
9	Ernst Fleißig	Mrz	6500	8125	1		Gesamtergebnis	35600	44500
10	Berta Emsig	Mrz	4000	5000	1				

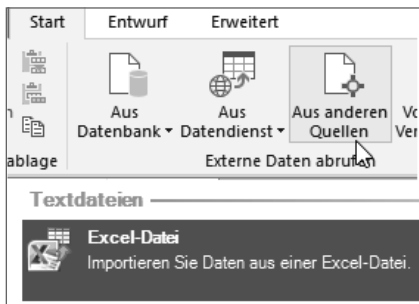
7.7.4 Berechnete Felder mit DAX in Power Pivot

Im Datenmodell werden Berechnungen in zusätzlichen Spalten hinterlegt, dazu kommen Formeln und DAX-Funktionen zum Einsatz. Arithmetische und logische Ausdrücke, Verweise und Relationen, Kontrollstrukturen und Schleifen stehen zur Verfügung. Alle Berechnungen und Beziehungen führen Sie im Power Pivot-Fenster durch.

Starten Sie im Register *Power Pivot* mit Klick auf *Datenmodell/Verwalten*.



Wählen Sie *Externe Daten abrufen/Aus anderen Quellen/Excel-Datei*.



Geben Sie der neuen Tabelle den Registernamen *Umsatz*.



Holen Sie die Liste aus der Datei und fügen Sie sie als Tabelle ein. Markieren Sie die nächste freie Spalte und wählen Sie im Kontextmenü der rechten Maustaste *Spalte umbenennen*.

	Name	Monat	Umsatz	
1	Hugo Tüchtig	Jan	2300	
2	Hugo Tüchtig	Feb	3200	
3	Hugo Tüchtig	Mrz	4100	
4	Ernst Fleißig	Jan	5000	
5	Ernst Fleißig	Feb	5500	

Tragen Sie diese DAX-Formel zur Berechnung der Provision ein:

f_x	=Umsatz[Umsatz]*1,25	
	Umsatz	Provision
	2300	2875
	3200	4000
	4100	5125
	5000	6250

```
=Umsatz[Umsatz]*1,25
```

Die Umsatzsumme und die Provisionssumme berechnen Sie als Measure im Berechnungsbereich. Schreiben Sie diese Formeln:

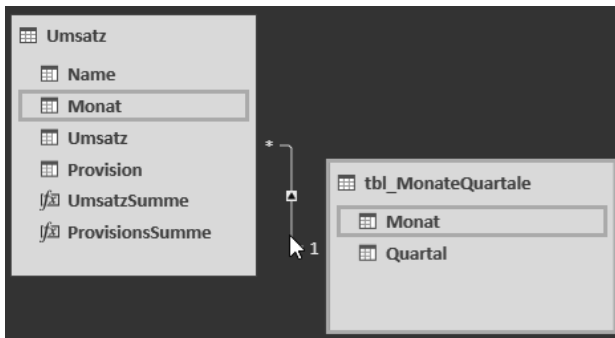
Externe Daten abrufen		
f_x	ProvisionsSumme:=sum(Umsatz[Provision])	
onat	Umsatz	Provision
n	2300	2875
b	3200	4000
z	4100	5125
UmsatzSumme: 35600		
ProvisionsSumme: 44500		

```
UmsatzSumme:=SUM(Umsatz[Umsatz])
ProvisionsSumme:=SUM(Umsatz[Provision])
```

Für die Quartalermittlung legen Sie in Excel eine Tabelle mit den Monaten und Quartalen an. Holen Sie diese als *tbl_QuartaleMonate* ins Datenmodell von Power Pivot.

Zu Datenmodell hinzufügen	Erkennen
Tabellen	Beziehungen
<div> <div>×</div> <div>✓</div> <div>f_x</div> </div>	
E	F
Monat	Quartal
Jan	1
Feb	1
Mrz	1
Apr	2
Mai	2

In der Diagrammansicht sehen Sie alle Tabellen im Datenmodell. Ziehen Sie eine Verknüpfungslinie von *Umsatz[Monat]* zu *tbl_MonateQuartale[Monat]*.

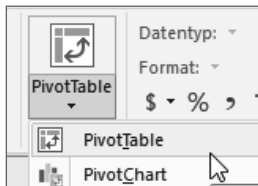


Mit der DAX-Funktion `RELATED()` erstellen Sie eine Verknüpfung auf das Feld *Quartal* in der Umsatztabelle. Nennen Sie die Spalte *Quartal*.

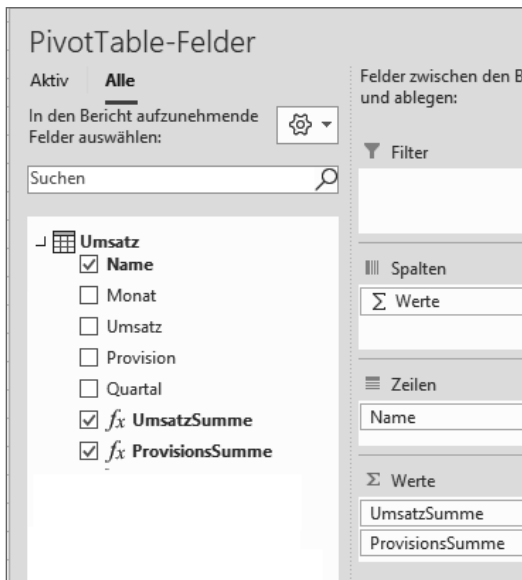
fx =RELATED(tbl_MonateQuartale[Quartal])			
	Umsatz	Provision	Quartal
	2300	2875	1
	3200	4000	1
	4100	5125	1
	5000	6250	1

```
=RELATED(tbl_MonateQuartale[Quartal])
```

Legen Sie mit Klick auf *PivotTable* eine neue PivotTable aus dem Datenmodell an.



Ziehen Sie das Feld *Name* in den Zeilenbereich und die beiden Measures *UmsatzSumme* und *ProvisionsSumme* in den Wertebereich. Das Feld *Quartal* wird wieder für einen Datenschnitt verwendet.



Index

Symbole

3D-Karten 373

A

Abrufen und Transformieren
49

Access 251

Active Directory 57

Azure 54, 255, 266

B

Berechnete Spalten

– Power Query 110

Bereichsnamen 386

Beziehungen 236

Beziehungsarten 13

Business Intelligence

– Anbieter 1

– Begriff 1

– mit Excel 2

– Werkzeuge 3

C

CSV 62, 463

– Power Pivot 279

Cube-Funktionen 344

D

Dashboard 475, 484

Daten abrufen und transformieren
53

Datenbanken

– Access 251

– Alternative Power BI 9

– für Power Query 54

– multidimensional 14

– relational 11

Datenimport 19, 48

Datenmodell 234

– anlegen 35

– Begriff 33

– explizit 38

– implizit 38

– in Power BI Desktop 461

– komprimieren 43

– Objekte 34

– PivotTable aus 303

– Regeln 33

Datenquellen

– für PivotTables 385

Datenschnitt 412

– Verbindungen 453

Datenüberprüfungsliste 124

DAX 230

– berechnete Felder 428

– Datumsberechnungen 287

– Datumstabelle 464

– Explizite Measures 332

– Funktionen 364

– Funktion erstellen 352

- Funktions-Assistent 354
- Funktionsreferenz 367
- Hilfe 366
- Kalendertabelle 339
- SUMX 333
- DAX-FunktionenKalender 466
- Dimensionen 14
- Dimensionsattribute 14
- Diskrete Anzahl 329
- Drilldown 126
- Dynamics 55

E

- Entity-Relationship-Modell 12
- ERM 12
- ETL 50, 457
 - Abkürzung 51
- Excel
 - laden nach Power BI 457
 - Mappe in Power BI veröffentlichen 477
- Excel 2013 34
- Explizite Measures 332
- Externe Daten
 - für Power BI Desktop 461
- Externe Daten abrufen 53

F

- Fehlzeitenanalyse 156
- Flächenkartogramm 369
- Formularsteuerelement 27

H

- Hadoop 57
- Hierarchien 316
 - in PivotTables 320

I

- IBCS 450
- IBM DB2 54
- Implizite Measures 239, 324
- Intelligente Tabelle 382, 387

J

- Joins 177
 - linker äußerer Join 177

K

- Karten
 - 3D 373
- Kartendiagramme 369
- KPIs 230, 339

L

- Liste 385
- Liste zur Tabelle 387

M

- M 135
 - Datumsberechnungen 147
 - Funktionsliste 146
 - Kalendertabelle 153
- Makroaufzeichnung 29
- M-Code 131
 - Dateipfad und Dateiname 132
 - Funktionsreferenz und Hilfe 146
 - Kommentare 143
 - Parameterabfrage 133
 - Syntax 139
 - Tabelle erstellen 425
- MDX 255
- MDX-Funktionen 257
- MDX-Referenzliste 259
- Measures 14, 324
 - in PivotTable erstellen 338
- M-Funktionen 113
 - Datum berechnen 115
 - Parameter 120
 - Spalten hinzufügen 114
 - Text kombinieren 116
- Microsoft Query 56
- MS Query 389
- MySQL 54

N

Normalform 12
Normalisierung 12
Nur Verbindung erstellen 128

O

OData 57
ODBC 19, 56, 389
ODBC-Datenquellen-Administrator 56
ODBC-Treiber 19, 23
ODBC-Verbindung 28
Office 365 34, 53
OLAP 243
OLAP-Cube 14
– Funktion 14
OLE DB 58
Online-Dienste 55
Oracle 54
Ordner
– auslesen mit Power Query 198

P

PBIX 463, 472
PivotCharts 30, 304, 378
– Altersstrukturanalyse 442
– aus Power Pivot 438
– Diagrammtypen 441
– Kennzahlencockpit mit 448
– mit Datenschnitt 448
– Prinzip 438
– Verbunddiagramm 451
PivotTable 30, 377
– aktualisieren 407
– Auswertbare Daten 383
– Bedingte Formatierung 406
– Berechnete Felder 416
– Bericht erstellen 394
– Berichtslayout 398
– Cube-Funktionen 344
– Datenmodelle 33, 35
– Datenquellen 385, 410
– Datenschnitte 304, 412
– Datenschnitteinstellungen 413

– Datumsfelder gruppieren 435
– Drilldown 411
– Feldliste 399
– Formate 403
– Gruppieren 432
– Layout als Standard speichern 403
– Layouts 400
– positionieren 395
– Prinzip 379
– Teilergebnisse 402
– Verbindungen 414
– verschieben 410
– Voraussetzungen 382
– Wertfeldeinstellungen 404
– Zahlenformate 406
– Zeitachse 415
PostgreSQL 54
Power BI
– Bericht veröffentlichen 474
– Cloud-Services 474
– Datenmodell 34
– Excel-Daten exportieren 479
– Excel-Daten veröffentlichen 476
– kombiniert mit Excel 457
– Konzept 455
– Vorstellung 6
Power BI Desktop 455, 458
– Daten importieren 461
– Datenmodell 466
– Datumstabelle 464
– filtern 473
– Komponenten 460
– PDF speichern 472
– Power Query 461
– Visualisierungen 467
Power BI Gateways 455
Power BI Mobile 455
Power BI oder Excel 8
Power BI Publisher 457, 481
Power BI Report Server 455
Power BI Services 455
Power Pivot
– Access 251
– Analysis Services 255
– Azure SQL 266
– Berechnungsbereich 232

- Beschreibungen 316
- Datenbanken 243
- Datenimport aus Azure 272
- Daten importieren 241
- Datenmodell 37, 393
- Datumstabelle 286
- DAX 350
- DAX-Funktionen 230
- Diagrammansicht 231
- Excel 276
- Geschichte 219
- Hierarchien 316
- KPIs 230, 339
- Measures 324
- Menüs 222
- Microsoft Analytic Platform 282
- OData 260
- Office-Versionen 220
- PivotTable 302, 428
- Spalten 309
- Spaltenformate 313
- Spezifikationen 219
- SQL-Abfragen 282
- starten 221
- Tabelleneigenschaften 285
- Tabellenregister 307
- Text/CSV 279
- Vorstellung 5
- Power Query 49
 - Abfrage bearbeiten 68
 - Abfrage-Editor 58, 69
 - Abfrage erstellen 61
 - Abfrage mit Pfad und Dateiname 122
 - Abfragen anfügen 98
 - Abfragen kombinieren 92
 - Abfragen transferieren 101
 - Abfrageoptionen 58
 - Abfrageschritte 85
 - Abfrage speichern 71
 - Abfragesprache 70
 - aus Datenbank 54
 - Ausfüllen 74
 - aus Tabelle/Bereich 390
 - Azure 107
 - Bearbeitungsleiste 136
 - Berechnete Felder 423
 - Berechnete Spalten 110
 - Dateien aus Ordern 198
 - Daten aus Ordner 53
 - Daten laden 58
 - Entpivotieren 82, 215
 - Erweiterter Editor 137
 - Excel 2010 52
 - Excel-Mappe 53
 - Gruppieren 78
 - im Menüband 49
 - installieren 52
 - Joins 177
 - M-Funktionen 113
 - ODC-Datei 105
 - Parameter 117
 - Praxis 156
 - Regionale Einstellungen 59
 - SAP-Berichte 211
 - Sicherheit und Datenschutz 59
 - Spalten trennen 74
 - Text/CSV 53
 - Transformationen 72
 - Transformieren 77
 - Typerkennung 177
 - Variable Spaltenüberschriften 196
 - Vorstellung 4
 - Webabfrage 168
 - Webabfragen 55
 - Zellwerte verwenden 185
- Power View 8
- Projekt Gemini 219

S

- Salesforce 55
- SAP-Berichte
 - mit Power Query aufbereiten 211
- SAP HANA 54
- Self Service Business Intelligence 1
- SharePoint 34, 55, 57f.
- SQL 20
 - mit Power Pivot 282
- SQL-Abfragen 21
- SQL-Anweisungen 20
- SQL-Befehl 26, 31
- SQL Server 243

SQL Server Analysis Services 243
SUCCESS 450

T

Tabelle 382
Tabelle aus Liste 387
Teradata 54

U

URL 55

V

VBA-Makro 28
– PivotTables aktualisieren 408
Verbunddiagramm 451
Verknüpfungen
– dynamische 19
Visualisierungen 467

Z

Zwischenablage 241