

Die Datenexplosion

Was ist End-User Business Intelligence?

Der Unterschied zwischen Unternehmens-BI  
und Self-Service BI

Wie hat sich Power BI entwickelt?

Was ist die Idee hinter Power BI?

# Kapitel 1

## Wie alles begann

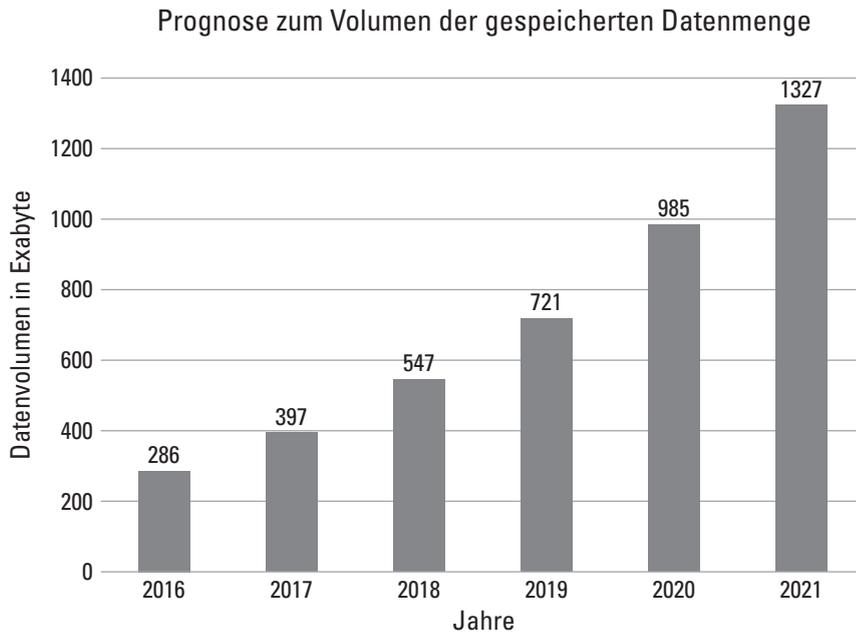
**W**ir leben in einer Welt, in der immer mehr Daten produziert werden. Inzwischen ist es denkbar, dass selbst Toaster Informationen über den Bräunungsgrad und die Zeitdauer des Röstens eines Toasts auf Facebook oder Twitter veröffentlichen. Und genau aus diesem Grund benötigt man Self-Service-BI-Tools, von denen übrigens das Beste Power BI ist. Wir sagen das nicht, weil wir über Power BI gerade ein Buch schreiben, sondern dieser Meinung ist auch das renommierte Analyseunternehmen Gartner, bei dem Power BI inzwischen mehrere Jahre hintereinander den besten Platz im sogenannten Gartner Magic Quadrant erzielt hat (<https://info.microsoft.com/ww-landing-gartner-mq-bi-analytics-2019.html?LCID=EN-US>).

## Die Datenexplosion

Schaut man sich Microsoft-Schätzungen zum weltweiten Datenaufkommen über die nächsten Jahre an, so kommt man zu den Zahlen in Abbildung 1.1.



Beachten Sie, dass es sich bei der Einheit in Abbildung 1.1 um *Exabyte* handelt. Ein Exabyte sind  $2^{60}$  Bytes, ausgeschrieben sind das 1.152.921.504.606.846.976 Bytes. Das entspricht einer Datenmenge von 1.691.556.350 oder knapp 1,7 Milliarden CDs.



**Abbildung 1.1:** Das geschätzte weltweite Datenaufkommen bis 2021

Daten: [de.statista.com](https://de.statista.com)

Was man an der Grafik in Abbildung 1.1 schön erkennen kann, ist, dass das Datenaufkommen exponentiell wächst. Aufgrund dieses exponentiellen Wachstums spricht man auch häufig von einer *Datenexplosion*. Man schätzt, dass sich die Datenmenge von 2016 bis 2021 verfünffachen wird. Die Zahlen stammen von Statista (<https://de.statista.com/statistik/daten/studie/819487/umfrage/prognose-zum-weltweit-gespeicherten-datenvolumen-in-rechenzentren/>).



Als *IoT* oder *Internet of Things* bezeichnet man die Vernetzung von Geräten über das Internet. IoT-Geräte lassen sich über das Internet steuern und erfassen Daten, die dann zur Speicherung und Weiterverarbeitung in Datenbanken im Internet weitergeleitet werden.

## Die drei Wellen der Business Intelligence

Unter *Business Intelligence* versteht man Techniken, mit denen man aus Daten, die auf unterschiedliche Art und Weise gesammelt werden, Informationen ableitet. Stellen Sie sich beispielsweise eine Online-Buchhandlung vor. Wenn dort Buchbestellungen eingehen, interessieren sich die Manager einer solchen Online-Buchhandlung nicht dafür, dass von einem bestimmten Benutzer das Buch »Power BI für Dummies« bestellt wurde, sondern eher dafür, wie oft »Power BI für Dummies« über einen gewissen Zeitraum bestellt wurde, um zu entscheiden, ob neue Bücher eingekauft werden müssen oder ob der aktuelle Lagerbestand noch ausreichend ist.

Dieses Beispiel zeigt sehr einfach, wie aus Daten Informationen abgeleitet werden können. Es werden Verkäufe aufsummiert und die Summe der verkauften Bücher über einen bestimmten Zeitraum wird mit den Büchern verglichen, die noch auf Lager sind. Aus diesem Vergleich ergibt sich die Zeitspanne, die die Bücher auf Lager noch ausreichen um die Kundenanforderungen – bei gleichbleibender Nachfrage – erfüllen zu können. Aus dieser Information kann der Zeitpunkt abgeleitet werden zu dem spätestens Bücher bestellt werden müssen damit die Online Buchhandlung nicht in einen Lieferengpass kommt. Informationen im Unternehmenskontext dienen immer dazu, aus diesen Informationen auch Handlungen abzuleiten die dann nicht »aus dem Bauch heraus« sondern auf Basis von nachvollziehbaren Fakten getroffen werden.

Schaut man sich die Geschichte der Business Intelligence über die Zeit an, so kann man drei Phasen oder Wellen ausmachen:

## **Erste Business-Intelligence-Welle – technische BI**

Die erste Business-Intelligence-Welle wird auch als *technische BI* bezeichnet und begann um 1996. Bei der technischen BI geht es darum, mit technischen Tools wie beispielsweise dem SQL Server, Reporting Services oder Analysis Services Business-Intelligence-Lösungen zu bauen. Dies war IT- oder Datenbankexperten vorbehalten, da diese die einzigen Personen waren, die die technischen Tools verstehen und anwenden konnten.

Der Prozess, eine Business-Intelligence-Lösung zu bauen, lief damals wie folgt ab: Der Anwender aus der Fachabteilung traf sich mit dem IT-Experten und erklärte ihm, welche Informationen und Berichte er benötigt. Der IT-Experte fing dann an, an der Lösung zu bauen. Dieses Vorgehen war problematisch, weil:

### ✓ **IT-Abteilungen haben wenig Zeit und sind überlastet.**

Wir kennen das alle: Wenn man in der IT-Abteilung anruft, dann sind dort schwer beschäftigte Menschen, die wenig Zeit haben, sich zusätzlich zu allen anderen Aufgaben auch noch mit der Erstellung von Business-Intelligence-Anwendungen zu beschäftigen. Das führte dazu, dass BI-Anforderungen nicht schnell umgesetzt werden konnten und es somit lange dauerte, bis eine Lösung bereitstand. Durch die Schnelligkeit der heutigen Zeit können sich in der Zwischenzeit die Auswertungsanforderungen geändert haben und die erstellte Lösung entspricht diesen nicht mehr.

### ✓ **IT-Abteilung und Fachanwender sprechen oft unterschiedliche Sprachen.**

Natürlich sprechen IT-Abteilung und Fachanwender in einem Unternehmen meistens dieselbe Muttersprache bzw. können sich auf jeden Fall aufs Englische zurückziehen. Das ist hier aber gar nicht gemeint. Denn: Fachabteilung und IT-Abteilung sprechen unterschiedliche Fachsprachen. Während einem Mitarbeiter der Fachabteilung möglicherweise völlig klar ist, wie man beispielsweise betriebswirtschaftliche Kennzahlen wie das EBIT berechnet, ist das in der IT-Abteilung vielleicht ein böhmisches Dorf. Während die IT-Abteilung wiederum sehr genau weiß, wie eine Datenbank funktioniert, sieht man beim Thema »Datenbank« in der Fachabteilung oft nur fragende Gesichter. Diese unterschiedlichen Know-how-Level führen dazu, dass es bei der Umsetzung der Business-Intelligence-Lösung große Reibungsverluste gibt.

✓ **Es werden mehrere Zyklen benötigt, bevor die Lösung fertig ist.**

Meist ist eine Business-Intelligence-Lösung nicht im ersten Wurf fertig, da während der Entwicklung neue interessante Informationen auftauchen oder neue Anforderungen an die Auswertung oder Visualisierung gestellt werden. Das ist ganz normal, da die Fachanwender gar nicht alle Möglichkeiten der Umsetzung der Lösung kennen und wenn diese erst einmal entdeckt werden, entsteht schnell weiterer Bedarf. Auch diese Tatsache führt dazu, dass sich Projekte verzögern.

Halten wir also fest: Das grundsätzliche Problem der technischen BI liegt darin, dass die Lösung von IT- oder Datenbankexperten umgesetzt werden muss und es relativ lange dauern kann, bis eine fertige Lösung bereitsteht. Diese Zeit für die Umsetzung hat man heute oft nicht mehr, sodass schnellere Vorgehensweisen gefunden werden müssen.

## Zweite Business-Intelligence-Welle – Self-Service BI

Bei der zweiten Business-Intelligence-Welle, der *Self-Service-BI-Welle*, die um 2008 begann, wurden von Herstellern BI-Tools angeboten, die von Analysten bedient werden konnten. Analysten sind im Unternehmen eher dazu in der Lage, fachliche Sachverhalte abzubilden, als IT-Experten, da sie täglich mit diesen Sachverhalten arbeiten, die zugrunde liegenden Daten kennen und auch deren Zusammenhänge verstehen.

So wundert es nicht, dass das erste Self-Service-BI-Tool, das Microsoft unter dem Namen Power Pivot herausgebracht hat, in Excel, dem Arbeitspferd der Analysten, Fachabteilungen und Controller, integriert wurde. Bei Power Pivot handelt es sich um eine Excel-Erweiterung, die dazu in der Lage ist, ein Datenmodell mit vielen Millionen Datensätzen abzubilden und auf einem Client Computer performant abzufragen.



Unter einem *Datenmodell* versteht man die Abbildung von Objekten der wirklichen Welt in sogenannten Entitäten in einer Datenbank und Beziehungen zwischen den Entitäten. Eine *Entität* ist die datentechnische Repräsentation eines Objektes wie beispielsweise eines Kunden oder einer Rechnung. Zwischen der Entität Kunde und der Entität Rechnung besteht eine Beziehung. Ein Kunde bekommt eine Rechnung. Das hier beschriebene Datenmodell wird auch als *Entity-Relationship-Modell* bezeichnet.

Schnell kamen drei weitere Plug-ins, Power Query, Power View und Power Map, für Excel heraus, die die Basis für Power BI bilden sollten. Die Idee war einfach: Weil man Excel als Basis für diese Produkte gewählt hatte, konnten nun auch Analysten ohne die Hilfe der IT-Abteilung Business-Intelligence-Auswertungen vornehmen.

## Dritte Business-Intelligence-Welle – End-User BI

Bei der dritten Welle der Business Intelligence, in der wir uns gegenwärtig befinden, spricht man auch von *End-User BI*. Jetzt soll jeder dazu in die Lage versetzt werden, Business-Intelligence-Auswertungen erstellen zu können. Dieser Anspruch ist recht hoch gegriffen, kann aber im Folgenden erklärt werden: Durch die Bereitstellung sehr einfacher Endanwender-Tools und vorgefertigter Analysen kann man den Kreis derer, die an Business

Intelligence arbeiten, wesentlich erweitern. Stellen Sie sich beispielsweise den Betreiber einer Website vor und nehmen wir einmal an, dass dieser keine Ahnung von Business Intelligence hat. End-User BI hilft ihm, mit fertigen Vorlagen die Google-Analytics-Daten seiner Website auszuwerten und aufzubereiten.

Dabei macht der Hersteller sich zunutze, dass jeder Website-Betreiber ähnliche Auswertungsanforderungen hat: Wie viele User besuchen meine Website? Wo kommen sie her? Welche Browser/Endgeräte werden benutzt? Wie lange halten sich die Besucher auf den einzelnen Webseiten auf? Und ähnliche Fragen.

Im Gegensatz zu einer Lösung aus der technischen BI der ersten Phase muss sich der Website-Betreiber aber nicht mit den vorhandenen Auswertungen zufriedengeben, sondern kann sie gemäß seinen eigenen Bedürfnissen erweitern und verändern.

## Ein Vergleich zwischen Unternehmens-BI und Self-Service BI

Wenn es um Self-Service- oder End-User-BI-Systeme geht, kommt oft auch der Begriff der *Unternehmens-BI* ins Spiel. In vielen Kundengesprächen, die wir geführt haben, wurde gesagt: »Wenn wir Power BI haben, brauchen wir ja dieses oder jenes Reporting-Tool gar nicht mehr!« Das ist nur teilweise richtig.

Grundsätzlich unterscheidet man zwischen Unternehmens-BI einerseits und Self-Service oder End-User BI andererseits. Bei der Unternehmens-BI handelt es sich um vorgefertigte Berichte, die genau definiert sind und die den Anwendern nur zum Lesen zur Verfügung gestellt werden. Unternehmens-BI-Berichte haben meist eine sehr klar vorgegebene Struktur und ein definiertes Aussehen. Ein gutes Beispiel für einen Unternehmens-BI-Bericht ist ein Jahresbericht, der Aktionären vorgelegt wird. Dieser ist in bestimmte Abschnitte gegliedert, wird produziert, damit sich die Aktionäre über die Geschäftstätigkeit des Unternehmens informieren können, und wird ausgedruckt und abgeheftet.



Unternehmens-BI-Berichte nennt man manchmal auch *Pixel Perfect*, weil diese Berichte ein sehr strenges Layout haben, das für den Druck optimiert ist und bei dem es darauf ankommt, dass jedes Element an der richtigen Stelle sitzt.

Das genaue Gegenteil der Unternehmens-BI-Berichte, die auch heute oft noch mit den Tools der ersten Welle der Business Intelligence erzeugt werden, stellen die Self-Service- oder End-User-BI-Berichte dar.

Bei dieser Berichtsart ist es entscheidend, dass der Anwender schnell an Informationen gelangt, indem er die Daten schnell in unterschiedlichsten Arten visualisieren oder miteinander verbinden kann. Hier geht es nicht um den reinen Konsum der Daten, sondern eher um Datenexploration. Der Anwender soll mit seinen Daten arbeiten und diese mal so oder mal so anzeigen können, um ein Verständnis für die Daten zu entwickeln.

Vor diesem Hintergrund ist verständlich, dass Power BI ein sehr interaktives Tool ist. Man kann auf die verschiedenen Elemente klicken und löst dadurch Filteraktionen aus, die sich auf die anderen Elemente im Bericht auswirken.



Eine häufige Frage, die wir immer wieder gestellt bekommen, ist: »Wo kann ich denn jetzt meinen Power-BI-Bericht ausdrucken?« Von Anfang an gab es bei Power BI nicht die Möglichkeit, direkt aus der Anwendung heraus zu drucken. Das hat auch einen guten Grund: Power BI lebt von der Interaktion des Benutzers mit den Daten. Druckt man einen Power-BI-Bericht aus, so verliert man diese Interaktionsmöglichkeit.

Über die verschiedenen Versionen von Power BI ist die Richtlinie, dass Power-BI-Berichte nicht ausgedruckt werden sollen, etwas aufgeweicht worden, da man inzwischen Power-BI-Berichte auch nach PowerPoint oder als PDF exportieren kann. Es macht aber viel mehr Sinn, Power-BI-Berichte am Rechner oder am Tablet zu betrachten.

## Die Entwicklung von Power BI

Die erste Version von Power BI wurde im Sommer 2010 von Amir Netz und Thierry D'Hers aus dem Microsoft-Reporting-Services-Team konzipiert und von Ron George entwickelt. Auf dem PASS Summit 2010, der weltgrößten Konferenz zum Thema Microsoft Data Platform, wurde Power BI unter dem Projektnamen Crescent vorgestellt. Ab dem 11. Juli 2011 konnte *Projekt Crescent* dann zusammen mit dem SQL Server Codename *Denali* heruntergeladen werden.

### Power-BI-Version 1.0

Unter dem eigentlichen Produktnamen wurde Power BI im September 2013 als *Power BI für Office 365* vorgestellt. Die erste Power-BI-Version bestand aus vier Excel-Plug-ins

### Power Pivot

*Power Pivot* stellt das Rückgrat von Power BI dar. Über Power Pivot kann man in Excel Datenmodelle mit vielen Millionen Datensätzen aufbauen. Dass man so unglaublich große Datenmengen mit Power BI verwalten kann, liegt daran, dass Power Pivot die Daten mit einer stark komprimierenden In-Memory-Technologie, die *Vertipaq* genannt wird, ablegt. Hierdurch kann man sehr große Datenmengen auf einem handelsüblichen PC verarbeiten.



Die grundlegende Idee hinter Vertipaq ist ziemlich trivial: Komprimierungsalgorithmen können umso besser komprimieren, je ähnlicher sich die zu komprimierenden Daten sind.

Schaut man sich einen herkömmlichen Datensatz an, zum Beispiel einen Eintrag in einer Adresstabelle, so sieht man, dass die Daten innerhalb eines Datensatzes sehr unterschiedlich sind (siehe Abbildung 1.2). Es gibt einen Namen, es gibt eine Postleitzahl, eine Adresse, eine Telefonnummer und so weiter. Da die Daten so unterschiedlich sind, können wir hier keine großen Komprimierungsraten erzielen.

Name	PLZ	Stadt	Adresse	Telefonnummer	Land
Frank Geisler	45329	Essen	Karlstraße 152	555-5555-555-1	Deutschland
Lieschen Müller	45329	Essen	Karlsplatz 1	555-5555-555-2	Deutschland
Hans Wurst	45330	Essen	Karlstraße 155	555-5555-555-3	Deutschland
Tillmann Eitelberg	53721	Siegburg	Am Turm 34	555-5555-555-4	Deutschland
Oliver Engels	65520	Bad Camberg	Otto-Hahn-Straße 22	555-5555-555-5	Deutschland
Wolfgang Strasser	4060	Leonding	Haidfeldstraße 3	555-5555-555-6	Österreich

**Abbildung 1.2:** Daten in einem Datensatz sind sehr unterschiedlich.

Anders sieht die Welt aber aus, wenn wir uns die gleiche Tabelle einmal spaltenweise anschauen (siehe Abbildung 1.3).

Name	PLZ	Stadt	Adresse	Telefonnummer	Land
Frank Geisler	45329	Essen	Karlstraße 152	555-5555-555-1	Deutschland
Lieschen Müller	45329	Essen	Karlsplatz 1	555-5555-555-2	Deutschland
Hans Wurst	45330	Essen	Karlstraße 155	555-5555-555-3	Deutschland
Tillmann Eitelberg	53721	Siegburg	Am Turm 34	555-5555-555-4	Deutschland
Oliver Engels	65520	Bad Camberg	Otto-Hahn-Straße 22	555-5555-555-5	Deutschland
Wolfgang Strasser	4060	Leonding	Haidfeldstraße 3	555-5555-555-6	Österreich

**Abbildung 1.3:** Daten in einem Feld sind ziemlich ähnlich.

Dann sind die Daten recht ähnlich und wir können eine sehr hohe Komprimierungsrate erzielen. Und genau das ist das Funktionsprinzip der Vertipaq Engine.

## Power Query

Dass man ein Datenmodell mithilfe von Power Pivot anlegen kann, ist schon mal eine sehr gute Sache – die Frage, die sich aber sofort stellt, ist, wie man man Daten in dieses Datenmodell bekommt. Um das Problem zu lösen, hat Microsoft Power Query entwickelt. Mithilfe von *Power Query* ist es möglich, Daten aus unterschiedlichsten Datenquellen in ein Power-Pivot-Datenmodell zu laden. Microsoft hat sich bei der Entwicklung von Power Query Mühe gegeben, die Unterschiede der Datenquellen für den Endanwender verschwinden zu lassen, sodass es für den Endanwender keinen großen Unterschied macht, ob er auf eine Textdatei, eine SQL-Server-Datenbank oder ein SAP-System zugreift. Der Zugriff von Power Query aus sieht immer ähnlich aus.



Auch wenn die Benutzeroberfläche von Power Query ziemlich einfach aussieht, passiert hinter den Kulissen wirkliche Magie. Power Query implementiert eine Technik, die sich *Query Folding* nennt und bei der geschaut wird, dass möglichst viel Last bei der Abfrageverarbeitung an die Backend-Datenquelle weitergegeben wird – sofern diese das unterstützt. Unterstützt die Datenquelle das nicht, bildet Power Query die fehlende Funktionalität ab.

Lassen Sie uns einmal ein Beispiel betrachten: Stellen Sie sich vor, dass aus einer Tabelle mit Rechnungen bestimmte Rechnungen herausgesucht und dann aggregiert werden sollen. Stellen Sie sich weiterhin vor, dass diese Tabelle zum

einen als CSV-Datei und zum anderen als SQL-Server-Tabelle vorliegt. Da eine CSV-Datei über keinerlei Möglichkeiten zum Filtern und Aggregieren verfügt – es ist ja nur eine einfache Textdatei –, werden die entsprechenden Funktionalitäten des Filterns *und* Aggregierens von Power Query übernommen. Anders sieht es bei der SQL-Server-Tabelle aus. Hier verfügt die relationale Datenbank-Engine, auf der der SQL Server aufgebaut ist, sehr wohl über Möglichkeiten zum Filtern und Aggregieren von Daten. Diese Funktionalitäten werden demnach von Power Query an den SQL Server abgegeben, der das viel performanter als Power Query ausführen kann.

Neben den Geschwindigkeitsvorteilen hat diese Vorgehensweise einen weiteren Vorteil – es werden nur die Daten an Power Query übergeben, die auch wirklich benötigt werden, und dadurch wird bei der Übertragung der Daten aus dem Internet Bandbreite gespart.

Stellen Sie sich vor, dass eine Million Zeilen zu einer einzigen Zahl aggregiert werden sollen. Sie können entweder eine Million Zeilen an Power Query übertragen und dort das Aggregat bilden lassen. Oder Sie lassen die Summe auf dem SQL Server berechnen und geben nur das Ergebnis an Power Query weiter.

Power Query ist Microsoft so gut gelungen, dass es vom Dasein eines Excel-Plug-ins unter vielen in der 2016er-Version von Excel zu einem festen Bestandteil befördert wurde. Vor diesem Hintergrund würde es uns auch nicht überraschen, wenn die regulären Möglichkeiten, Daten in Excel zu laden, verschwinden und der Zugriff auf externe Datenquellen nur noch über Power Query stattfinden würde.

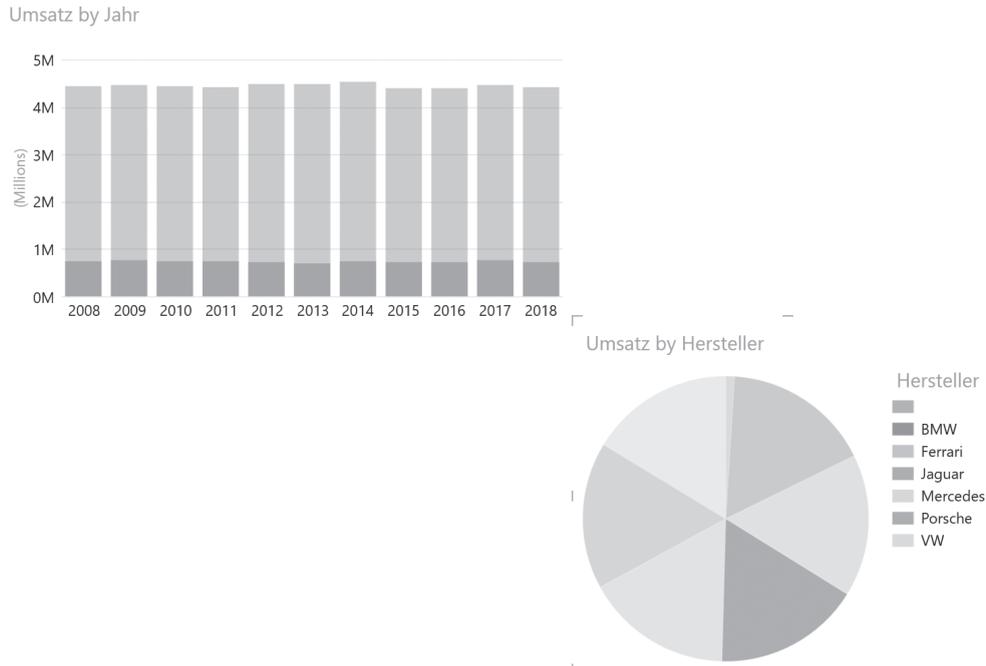
## Power View

Nachdem es gute Möglichkeiten gab, Datenmodelle in Power Pivot anzulegen und über Power Query zu befüllen, fehlten nun Möglichkeiten, die Daten darzustellen. Hierzu wurde *Power View* von Microsoft entwickelt, ein weiteres Excel-Plug-in, mit dem man interaktive Diagramme erstellen konnte.

Natürlich kann man die Daten, die mit Power Query geladen und mit Power Pivot aufbereitet wurden, unter Excel beispielsweise als Pivot-Tabelle anzeigen lassen. Diese Darstellungsweise hat aber den Nachteil, dass es keine Möglichkeit gibt, interaktiv mit den Daten zu arbeiten, genau wie das auch mit normalen Excel-Diagrammen nicht möglich ist.

Power View bietet bereits die später viel weiter ausgebaute Möglichkeit, mit den Daten interaktiv zu arbeiten. So kann man auf Elemente einer Grafik klicken und dieser Klick sorgt dafür, dass andere Diagramme des Berichts entsprechend gefiltert werden. Werden in einem Bericht in dem einen Diagramm Automarken und in dem anderen Diagramm Jahresumsätze gezeigt, so kann man über einen Mausklick auf eine bestimmte Automarke anzeigen, wie hoch der Anteil dieser Marke an den jeweiligen Jahresumsätzen war (siehe Abbildung 1.4).

## Übersicht KFZ Umsätze



**Abbildung 1.4:** Mit dem Kreisdiagramm wird das Balkendiagramm gefiltert.

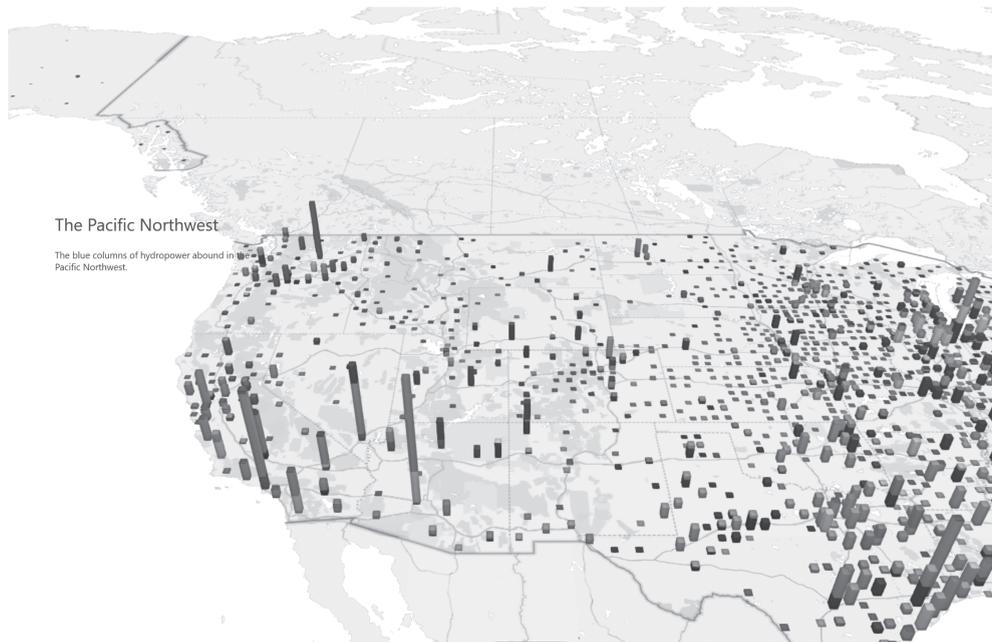
## Power Map

Oft müssen in Unternehmen Geodaten visualisiert werden. Hierfür hat Microsoft *Power Map* entwickelt, ein Excel-Plug-in, welches die Darstellungen von Daten mit Geobezug ermöglicht (Abbildung 1.5). Die Daten für die Geocodierung kommen vom Bing-Maps-Service, sodass selbst eine Darstellung der Daten auf Hausnummernebene möglich ist.



Unter *Geocodierung* versteht man die Übersetzung von Adressdaten in Längen- und Breitengrade. Positionen auf einer Karte werden über Längen- und Breitengrade angegeben. Möchten Sie Adressdaten auf einer Karte darstellen, so geht das nicht direkt, sondern die Adresse muss zunächst in das Koordinatensystem der Karte, also Längen- und Breitengrade, umgerechnet werden.

Es können viele unterschiedliche Themes für die Darstellung der Karte gewählt werden und neben der Möglichkeit, die Datendarstellung auf der Karte über die Zeit zu animieren, ist es auch möglich, sogenannte Rundflüge oder Touren zu definieren, bei denen die 3D-Kamera über die Karte bewegt wird und es somit möglich ist, die räumliche Datenverteilung auf der Karte noch besser zu verstehen.



**Abbildung 1.5:** Darstellung von Kraftwerken auf einer 3D-Karte mit Power Map

## Was noch?

Neben diesen lokal installierten Excel-Plug-ins gehörten auch schon zur ersten Version von Power BI Cloud-Dienste, die unter Microsoft Azure zur Verfügung gestellt wurden und eine sehr starke Integration in Office 365 hatten.

Die Cloud-Dienste dienten hauptsächlich der Zusammenarbeit mit Power BI im Team und sind in der Version 2.0 von Power BI wesentlich erweitert worden. Die starke Bindung der Dienste zu Office 365 ist aufgehoben worden. Da die Cloud-Dienste der ersten Power-BI-Version heute nicht mehr zur Verfügung stehen, gehen wir in diesem Buch nicht mehr weiter auf sie ein.

## Probleme

Power BI in der Version 1.0 war vom Umfang und den technischen Möglichkeiten schon ziemlich gut. Was aber nicht so gut war, war die Implementierung in Excel. Strategisch war es sicherlich nicht verkehrt, das Microsoft Self-Service-Business-Intelligence-Tool in Excel einzubauen, das Produkt, das jeder Controller und jeder, der im Unternehmen mit Zahlen arbeitet, kennt. Dieser Ansatz hatte aber einen entscheidenden Nachteil – die Implementierung in Excel war nicht besonders gut und hat für viel Verwirrung gesorgt.

Dass die vier Tools der Power BI in Excel implementiert wurden und dass dies auch noch auf unterschiedliche Art und Weise passierte, hat für viel Verwirrung gesorgt. Manche Plug-ins, wie Power Map, haben lange Zeit ihren Preview-Status nicht verlassen. Power Pivot war schon Bestandteil von Excel 2010, Power View kam in Excel 2013 hinzu und Power Query und Power Map fanden dann in Excel 2016 den Weg in den Lieferumfang von Excel. Power

View ist allerdings schon wieder auf dem Weg nach draußen, da es standardmäßig in Excel 2016 nicht mehr installiert ist und man einige Klimmzüge durchführen muss, um Power View in Excel 2016 zu aktivieren. Das ist insofern nachvollziehbar, als dass Power View auf *Silverlight* basiert, einer von Microsoft abgekündigten Technologie.

Aus Anwendersicht ist das alles natürlich ein Wartungs- und Installationsalptraum. Welche Excel-Version habe ich? Welche Power-BI-Tools sind schon enthalten und müssen lediglich aktiviert werden und welche Power-BI-Tools müssen aus dem Internet heruntergeladen und installiert werden? Welche Komponente muss wann aktualisiert werden?

## Power BI 2.0

Vor diesem Hintergrund und vor dem Hintergrund der anderen Self-Service-BI-Tools am Markt wie *Tableau* oder *Qlikview* hat man sich bei Microsoft überlegt, dass es wahrscheinlich schlauer ist, eine monolithische Anwendung zu erstellen, in der die Komponenten Power Pivot, Power Query und Power View miteinander vereinigt werden. Das war dann die Geburtsstunde von *Power BI Desktop*, der Anwendung, mit der man Power-BI-Berichte erstellen und in der Cloud veröffentlichen kann.



Power Map hat zum Zeitpunkt, zu dem wir dieses Buch schreiben, leider noch nicht den Weg nach Power BI gefunden. Es gibt zwar Custom Visuals, also benutzerdefinierte Diagrammtypen, die den Look von Power Map nachahmen, allerdings ohne dass sie den entsprechenden Funktionsumfang besäßen.

Die zweite Version von Power BI wurde auf der WPC (World Partner Conference) 2015 vorgestellt und das ist auch die Version, mit der wir uns in diesem Buch ausschließlich beschäftigen.

Power BI 2.0 besteht aus zwei Hauptkomponenten: Zum einen ist das Power BI Desktop, das man sich lokal herunterladen kann. Zum anderen sind das die Cloud Services, also die Dienste, die bei Microsoft im Rechenzentrum laufen und die unter anderem dafür sorgen, dass man Power-BI-Berichte mit anderen Personen über einen Webbrowser teilen kann.



Power BI Desktop ist ein Produkt, an dem Microsoft sehr intensiv arbeitet. Jeden Monat – ja, Sie haben richtig gelesen: jeden Monat – kommt eine neue Version von Power BI Desktop heraus. Diese neue Version bedeutet kein komplettes Redesign, wie es zwischen der Power BI 1.0 und Power BI 2.0 stattgefunden hat, sondern ist lediglich eine erweiterte Version mit neuen Funktionalitäten und Fehlerbehebungen. Das grundlegende Bedienkonzept, das Sie in diesem Buch kennenlernen, bleibt weitestgehend unverändert.

In einer Unternehmensumgebung, in der Anwendungen zur Verteilung auf die Client-Rechner paketiert werden müssen, ist das natürlich schon eine ziemliche Herausforderung, jeden Monat ein neues Power-BI-Paket ausrollen zu müssen. Daher gibt es in der Kombination mit Windows 10 eine bessere Möglichkeit, Power BI zu installieren: Installieren Sie Power BI Desktop – Achtung: Achten Sie genau auf den Namen, da Verwechslungsgefahr mit einer suboptimalen

Power-BI-App besteht – einfach über den *Microsoft Store*. Genau wie jede andere App, die aus dem Microsoft Store installiert wurde, wird Power BI dann immer aktualisiert, wenn eine neue Version verfügbar ist. Das funktioniert genauso wie auf Ihrem Smartphone.



Wenn Sie mit Power BI im Team arbeiten, müssen Sie dafür sorgen, dass alle Teammitglieder ihre Power-BI-Versionen gleichzeitig (oder nahezu gleichzeitig) aktualisieren, da man eine Power-BI-Datei, die mit einer neueren Version von Power BI Desktop abgespeichert wurde, nicht mit einer älteren Power-BI-Desktop-Version öffnen kann.

## Die Power Platform

Während wir dieses Buch schreiben, etabliert Microsoft gerade die sogenannte *Power Platform*. Power BI – das Produkt, von dem dieses Buch handelt – ist ein hervorragendes Tool, um Daten zu laden, Daten zu bearbeiten und Daten zu visualisieren. Es ist in der Bedienung sehr einfach, sodass man kein IT-Experte sein muss, um mit Power BI ansprechende Auswertungen zu erstellen.

Power BI ist aber auf die reine Analyse und Visualisierung von Daten beschränkt. Man hat keine Möglichkeit, über Power BI Daten einzugeben oder Prozesse zu automatisieren. Und das ist genau der Punkt, wo nun die Power Plattform ins Spiel kommt.

Microsoft hat Power BI zwei weitere Tools an die Seite gestellt, die zusammen mit Power BI die Power Platform bilden: Power Apps und Microsoft Flow.

### ✓ Power Apps

Mithilfe von *Power Apps* lassen sich recht einfach Dateneingabeanwendungen erstellen und die eingegebenen Daten in beliebige Datenquellen schreiben. So ist es egal, ob die Daten in eine SQL-Server-Datenbank, einen Oracle Server, eine Textdatei oder gar in eine Excel-Datei geschrieben werden. Die Anwendungen, die man mithilfe von Power Apps schreiben kann, beschränken sich auf Anwendungen zur Datenerfassung. So können Sie mit Power Apps weder Word nachprogrammieren noch ein Computerspiel schreiben.

Der Clou bei Power Apps ist aber, dass es nicht nur im Browser auf dem Laptop oder PC ausgeführt werden kann, sondern dass es eine Power-App-Anwendung auch für die gängigen Smartphone-Betriebssysteme iOS und Android gibt (ja, es gibt auch eine Windows Phone App, aber da das Windows-Phone-Betriebssystem von Microsoft abgekündigt ist, betrachten wir das an dieser Stelle nicht weiter). Innerhalb der Smartphone-App können alle Sensoren des Smartphones abgefragt werden, sodass man beispielsweise mit der Kamera des Telefons Bilder aufnehmen kann, die dann in der Datenbank gespeichert werden, oder die GPS-Koordinaten des Smartphones abfragen und speichern kann.



Auch wenn Microsoft behauptet, dass es mit Power Apps ohne Programmierung möglich ist, Apps zu entwickeln, können wir dieser Aussage hier nur so halb zustimmen. Richtig ist, dass es viel einfacher ist, mit Power Apps mobile Anwendungen zu erstellen, als das bisher mit klassischen Programmiersprachen

wie beispielsweise C# möglich war. Fakt ist aber auch, dass man, um bestimmte Funktionen und Effekte zu erzielen, diese in einer Excel-ähnlichen Programmiersprache beschreiben muss.

### ✓ Microsoft Flow

*Microsoft Flow* ist eine Anwendung, mit der man Abläufe automatisieren kann. In Vorträgen und auf Konferenzen umschreiben wir Microsoft Flow gerne als »Wenn-dann-Engine«. Wenn ein bestimmter Zustand erreicht ist oder wenn ein bestimmtes Ereignis eingetroffen ist, dann führe automatisch eine bestimmte Aktion aus.

Hier ein simples Beispiel für solch einen Ablauf: Wenn jemand etwas über Power BI für Dummies twittert, dann soll dieser Tweet automatisch in eine Excel-Tabelle geschrieben werden und außerdem soll der Twitter-User automatisch im CRM-System (Customer Relationship Management) von Wiley angelegt werden.

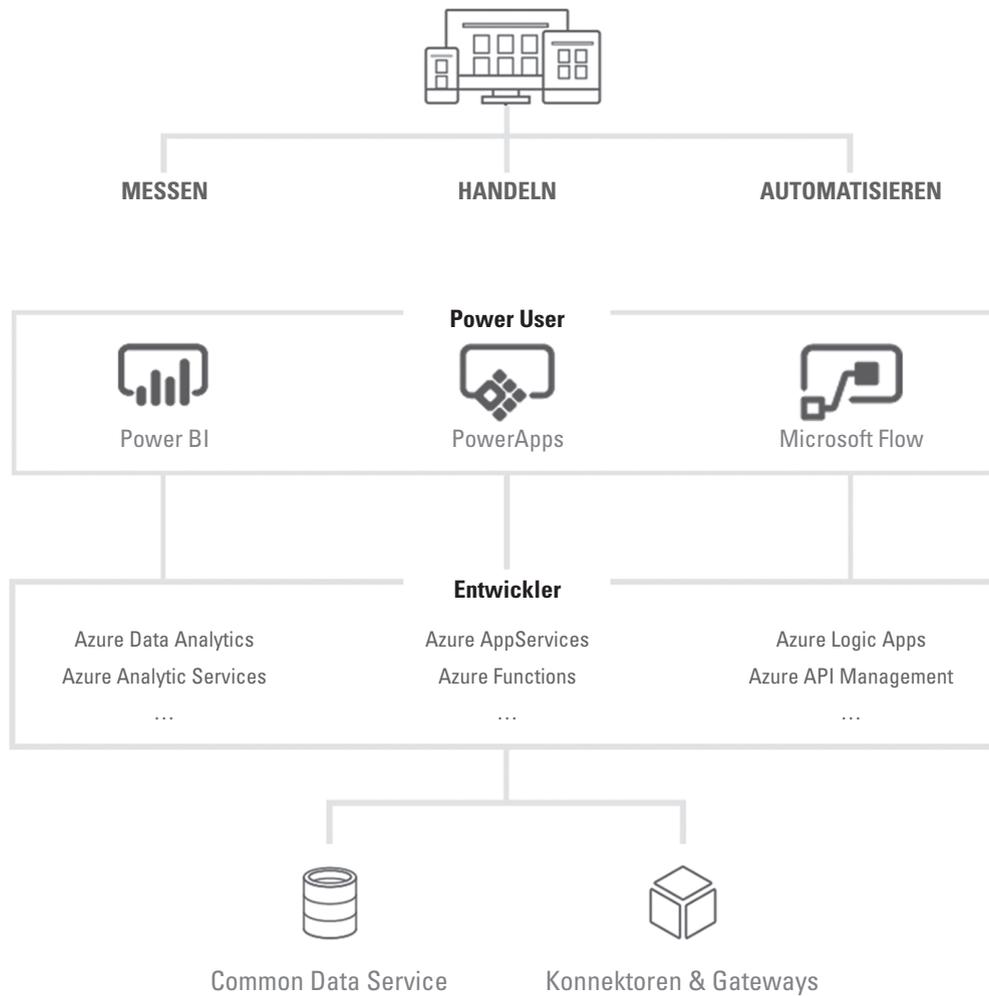
Der auslösende Trigger in diesem Beispiel ist ein Tweet, der entweder das Hashtag #powerbifürdummies oder einen Bezug auf den Twitter-User @WileyVCH enthält. Die Aktionen, die dann ausgelöst werden – es kann davon mehrere geben –, sind zum einen, dass der Tweet samt User und Text in einer Excel-Datei gespeichert wird, und zum anderen, dass der Twitter-User als Kontakt im CRM-System angelegt wird.

Microsoft Flow kommuniziert mit anderen Anwendungen über sogenannte *Konnektoren*. Inzwischen gibt es eine Vielzahl an Konnektoren zu den meisten häufig eingesetzten Systemen.

In Abbildung 1.6 sehen Sie eine schematische Skizze der Power Platform. Auf dieser Basis kann man sehr einfach angepasste Businessanwendungen entwickeln. Bei den Businessanwendungen handelt es sich nicht um komplexe Systeme wie SAP oder Microsoft Dynamics, sondern um einfache Anwendungen, mit denen beispielsweise Daten auf Mobilgeräten gesammelt und dann per Power BI ausgewertet werden können. Über die zur Verfügung stehenden Konnektoren und Gateways können auch bestehende Anwendungen erweitert werden, sodass man relativ einfach eine mobile Oberfläche für einen bestimmten Aspekt eines bestehenden Systems erstellen kann.

Um dies zu verdeutlichen, schauen wir uns einfach mal ein mittelständisches Unternehmen an, welches ein ERP-System (Enterprise Resource Planning) einsetzt, in dem unter anderem der Lagerbestand der vom Unternehmen hergestellten Produkte verwaltet wird. Stellen wir uns weiter vor, dass es bisher keinen digitalen Prozess zur Lagerinventur gibt. In unserem Szenario müssen die Mitarbeiter, welche die Lagerinventur durchführen, alle Ergebnisse auf einen Zettel schreiben und die so erhobenen Informationen später am Rechner in das System einpflegen. Dieser Prozess ist natürlich suboptimal, da doppelt gearbeitet werden muss. Erst werden die Informationen erfasst und dann müssen sie auch noch in den Rechner übertragen werden. Ein mühsamer und fehleranfälliger Prozess.

Mit der Power Platform und vor allem mit Power Apps ist es recht einfach möglich, eine Oberfläche für Smartphones oder Tablets zu entwickeln, mit denen die Informationen direkt während der Erhebung digital erfasst werden können. Basiert das ERP-System auf einer Standarddatenbank und sind die Schnittstellen zum System dokumentiert und offengelegt, ist es sogar möglich, die mit der Power App erfassten Informationen direkt in das System zu übertragen.



**Abbildung 1.6:** Die Power Plattform

Neben den Tools für den Fachanwender – Power BI, Power Apps und Microsoft Flow –, die allesamt auf Azure-Technologien wie *Azure Functions* oder *Azure Logic Apps* aufbauen, gibt es für Entwickler die Möglichkeit, die Power Plattform zu erweitern. Werden ganz spezielle Funktionalitäten gefordert, die die Plattform so nicht unterstützt, ist es jederzeit möglich, diese Funktionalitäten auf Basis der zugrunde liegenden Azure-Dienste zu erstellen.