

IBM DB2 Cube Views



Geschäftsmodellszenarios

Version 8

IBM DB2 Cube Views



Geschäftsmodellszenarios

Version 8

Hinweis

Hinweis: Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 33 gelesen werden.

- Die IBM Homepage finden Sie im Internet unter: **ibm.com**
- IBM und das IBM Logo sind eingetragene Marken der International Business Machines Corporation.
- Das e-business-Symbol ist eine Marke der International Business Machines Corporation.
- Infoprint ist eine eingetragene Marke der IBM.
- ActionMedia, LANDesk, MMX, Pentium und ProShare sind Marken der Intel Corporation in den USA und/oder anderen Ländern.
- C-bus ist eine Marke der Corollary, Inc. in den USA und/oder anderen Ländern.
- Java und alle auf Java basierenden Marken und Logos sind Marken der Sun Microsystems, Inc. in den USA und/oder anderen Ländern.
- Microsoft Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.
- PC Direct ist eine Marke der Ziff Communications Company in den USA und/oder anderen Ländern.
- SET und das SET-Logo sind Marken der SET Secure Electronic Transaction LLC.
- UNIX ist eine eingetragene Marke der Open Group in den USA und/oder anderen Ländern.
- Marken anderer Unternehmen/Hersteller werden anerkannt.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM DB2 Cube Views Business Modeling Scenarios, Version 8,
IBM Form SC18-7803-00,
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2003
© Copyright IBM Deutschland GmbH 2003

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
SW TSC Germany
Kst. 2877
Juni 2003

Inhaltsverzeichnis

Zu diesem Handbuch	v	Kapitel 4. Zählen der Internetbestellungen	21
Zielgruppe	v	Details des Szenarios	21
Kapitel 1. Berechnen von Fluktuation und Wert des Lagerbestands in einem Warenlager im Zeitverlauf	1	Schritte zum Erstellen des Maßes	22
Details des Szenarios	1	Kapitel 5. Erstellen einer Rangfolge der Verkaufszahlen	25
Schritte zum Erstellen von Maßen	4	Details des Szenarios	25
Kapitel 2. Korrelieren von Werbekosten und Verkaufszahlen	9	Schritte zum Erstellen des Maßes	27
Details des Szenarios	9	Kapitel 6. Erstellen einer Dimension 'Zeit' anhand von Zeitdaten aus der Fakttabelle	29
Schritte zum Erstellen des Maßes	9	Details des Szenarios	29
Kapitel 3. Berechnen des Gewinns und der Gewinnspanne einer Filiale	15	Schritte zum Erstellen der Attribute und der Dimension	31
Details des Szenarios	15	Bemerkungen	33
Schritte zum Erstellen der Maße	16	Marken	35

Zu diesem Handbuch

Dieses Buch enthält ergänzende Informationen für das Handbuch *DB2 Cube Views Installation und Benutzerhandbuch* zum Modellieren häufig vorkommender realistischer Geschäftsszenarios mit Metadaten von DB2 Cube Views.

Zielgruppe

Sie sollten dieses Buch lesen, wenn Sie als Datenbankadministrator (DBA) mit OLAP-Metadaten und DB2 Universal Database (DB2) arbeiten. Dabei sollten Sie mit den folgenden Dingen vertraut sein:

- OLAP-Konzepte (zum Beispiel Sternschemata)
- Metadatenobjekte in DB2 Cube Views (zum Beispiel Kubusmodelle, Faktenobjekte, Dimensionen, Verknüpfungen, Maße und Attribute)

Kapitel 1. Berechnen von Fluktuation und Wert des Lagerbestands in einem Warenlager im Zeitverlauf

Ein Einzelhandelsunternehmen XYZ lagert seinen Warenbestand in einem Warenlager, bevor die Waren zum Verkauf in eine Filiale gebracht werden. Das Einzelhandelsunternehmen XYZ erfasst zeitbezogene Daten zum Status des Lagerbestands im Warenlager und möchte diese Daten analysieren. Dabei sollen speziell die beiden folgenden Aspekte des Lagerbestands im Warenlager untersucht werden:

- Der Warenfluss in das und aus dem Warenlager
- Der Wert des Warenbestands im Warenlager zu einem gegebenen Zeitpunkt

Zum ersten Aspekt, dem Warenfluss, gehört das Untersuchen von Daten im Zeitverlauf. Der zweite Aspekt, der Wert des Lagerbestands, ist eine Momentaufnahme des Warenbestands im Warenlager zu einem bestimmten Zeitpunkt.

Details des Szenarios

Das Einzelhandelsunternehmen XYZ verfügt über eine Faktentabelle mit folgenden auf das Warenlager bezogenen Spalten: QUANTITY_IN, QUANTITY_OUT, CURRENT_QUANTITY, PRODUCT_VALUE, PRODUCT_ID und TIME_ID. Diese Daten werden wöchentlich in die Tabelle eingegeben. Die Datenbank verfügt außerdem über eine Produkttabelle und eine Zeittabelle. Beispielsweise zeigt Tabelle 1 eine Stichprobe der Daten aus der Faktentabelle.

Tabelle 1.

PRODUCT_ID	TIME_ID	QUANTITY_IN	QUANTITY_OUT	CURRENT_QUANTITY	PRODUCT_VALUE
1234	1	5	0	5	5
1234	2	20	10	15	5
1234	3	10	20	5	5

Der Wert für PRODUCT_ID der drei Beispieldatensätze ist gleich, weil jeder Produkttyp mehrmals in das Warenlager gebracht und wieder daraus entnommen werden kann.

Der DBA des Einzelhandelsunternehmens XYZ muss drei verschiedene Maße erstellen:

Flow In

Modelliert den Warenfluss in das Warenlager.

Flow Out

Modelliert den Datenfluss aus dem Warenlager.

Current Value

Modelliert den Wert des Warenbestands zu einem gegebenen Zeitpunkt.

Zum Erstellen der ersten beiden Maße **Flow In** und **Flow Out** erstellt der DBA Maße, die auf die Spalten QUANTITY_IN bzw. QUANTITY_OUT abgebildet werden und die Daten für alle Dimensionen kumulieren. Dies wird auch als vollständig additives Maß bezeichnet, weil die Daten nur mit der Funktion SUM für alle Dimensionen zusammengefasst werden. Beispielsweise zeigt Tabelle 2 eine Gruppe von Beispieldaten für die Spalten QUANTITY_IN und QUANTITY_OUT über drei Monate für das Produkt mit der PRODUCT_ID (Produkt-ID) 1234. Die Maße **Flow In** und **Flow Out** summieren diese monatlichen Werte und berechnen die Gesamtmengen, die innerhalb des Quartals in das Warenlager gebracht und daraus entnommen wurden.

Tabelle 2. Berechnung der Beispieldaten für die vollständig additiven Maße 'Flow In' und 'Flow Out' für PRODUCT_ID 1234

	Januar	Februar	März	1. Quartal
QUANTITY_IN	5	20	10	35
QUANTITY_OUT	0	10	20	30

Vollständig additive Maße sind die einfachsten und gängigsten Maße, und sie werden häufig als Bausteine für komplexere Maße verwendet. Für Maße, die auf numerischen Quelldaten basieren, erstellt die OLAP-Zentrale standardmäßig ein vollständig additives Maß.

Zum Erstellen des dritten Maßes (Current Value = aktueller Wert) erstellt der DBA ein berechnetes Maß, das den Wert durch Multiplizieren von PRODUCT_VALUE mit CURRENT_QUANTITY ermittelt. Beispiel: Für ein Produkt mit PRODUCT_ID=1234 und dem Wert 5 ist das Maß **Current Value** in Tabelle 3 angegeben.

Tabelle 3. Berechnung der Beispieldaten für das Maß Current Value für PRODUCT_ID 1234

	Januar	Februar	März
CURRENT_QUANTITY	5	10	20
Current Value	25	50	100

Diese Daten müssen für alle Dimensionen zusammengefasst werden. Weil dieses Maß jedoch den Wert zu einem bestimmten Zeitpunkt errechnet, ist es sinnlos, die Summe für die gesamte Dimension **Time** zu ermitteln.

Statt dessen werden bei der Spaltenberechnung die Daten für die Dimension **Product** summiert und ein Mittelwert der Daten im Zeitverlauf gebildet. Dies wird als halbadditives Maß bezeichnet, weil nur ein Teil der Spaltenberechnung die Funktion SUM beinhaltet.

Maße, die Daten für Momentaufnahmen berechnen (Daten, die einen bestimmten Zeitpunkt darstellen), zum Beispiel monatliche Bestandsdaten, sind häufig halbadditive Maße, weil es nicht sinnvoll wäre, die Monate zu Quartalen zusammenzufassen. Wenn ein Produkt ein ganzes Quartal in einem Warenlager verbleibt, wird dieses Produkt an allen drei Monaten des Quartals in den Momentaufnahmedaten für CURRENT_QUANTITY des Warenlagerinventars erfasst. Wenn die Daten für CURRENT_QUANTITY im Zeitverlauf summiert werden, wird ein Produkt, das drei Monate lang im Warenlager liegt, drei Mal gezählt. Wie in Tabelle 4 gezeigt, hat der Wert 25 für das 1. Quartal keine Bedeutung für die Vorgänge im Warenlager. Die Tabelle zeigt, dass sich zu keinem Zeitpunkt 25 Produkte im Warenlager befunden haben. Es ist also sinnlos, den Wert von 25 Produkten zu berechnen.

Tabelle 4. Berechnung der Beispieldaten für die Spalte CURRENT_QUANTITY mit der Funktion SUM in der Zeitdimension für PRODUCT_ID 1234

	Januar	Februar	März	1. Quartal
SUM(CURRENT_QUANTITY)	5	15	5	25

Anstatt die Funktion SUM auf alle Dimensionen anzuwenden, können Sie andere Spaltenberechnungsfunktionen wie AVG, MIN oder MAX für die Zeitdimension ausführen. Beispielsweise können Sie auf die gleiche Gruppe von Beispieldaten für Januar, Februar und März eine zweite Spaltenberechnungsfunktion für die Zeitdimension anwenden, wie in Tabelle 5 gezeigt, um sinnvolle Werte für das Quartal zu erhalten. Das Maß **Current Value** kann den durchschnittlichen Gesamtwert des während des Quartals in dem Warenlager vorrätigen Lagerbestands liefern bzw. den Minimal- oder Maximalwert zu einem beliebigen Zeitpunkt des Quartals.

Tabelle 5. Berechnung der Beispieldaten für die Spalte CURRENT_QUANTITY mit den Funktionen AVG, MAX und MIN in der Zeitdimension für PRODUCT_ID 1234

	Januar	Februar	März	1. Quartal
AVG(CURRENT_QUANTITY)	5	15	5	8,3
MAX(CURRENT_QUANTITY)	5	15	5	15
MIN(CURRENT_QUANTITY)	5	15	5	5

Schritte zum Erstellen von Maßen

Die folgenden Schritte zeigen, wie Sie mit dem Fenster **Faktenmerkmale** der OLAP-Zentrale die Maße **Flow In**, **Flow Out** und **Current Value** in einem vorhandenen Faktenobjekt erstellen können:

1. Klicken Sie das Faktenobjekt in der Objektbaumstruktur der OLAP-Zentrale mit der rechten Maustaste an, und klicken Sie **Maße editieren** an. Das Fenster **Faktenmerkmale** wird geöffnet.
2. Gehen Sie wie folgt vor, um das Maß **Flow In** zu erstellen:
 - a. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnetes Maß erstellen** an, um das Maß **Flow In** zu erstellen. Das Fenster **SQL Expression Builder** wird geöffnet.
 - b. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung **FLOW IN** in das Feld **Name** ein.
 - c. Führen Sie die folgenden Schritte aus, um den SQL-Ausdruck 'Flow In' zu erstellen:
 - Erweitern Sie den Ordner **Spalten** und die Fakttable in der Liste **Daten**.
 - Klicken Sie die Spalte **QUANTITY_IN** doppelt an, um Sie dem Ausdruck hinzuzufügen.
 - Klicken Sie **OK** an, um das Fenster **SQL Expression Builder** zu schließen. Sie brauchen die Standard-Spaltenberechnungsfunktion **SUM** auf der Seite **Spaltenberechnungen** nicht zu ändern. Die Funktion **SUM** ist die Standardeinstellung für das Maß **Flow In**, weil die Datenquelle numerisch ist und das Maß auf eine Spalte verweist und nicht nur auf andere vorhandene Maße.
3. Gehen Sie wie folgt vor, um das Maß **Flow Out** zu erstellen:
 - a. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnete Maße erstellen** an, um das Maß **Flow Out** zu erstellen. Das Fenster **SQL Expression Builder** wird geöffnet.
 - b. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung **FLOW OUT** in das Feld **Name** ein.
 - c. Führen Sie die folgenden Schritte aus, um den SQL-Ausdruck 'Flow Out' zu erstellen:
 - Erweitern Sie den Ordner **Spalten** und die Fakttable in der Liste **Daten**.
 - Klicken Sie die Spalte **QUANTITY_OUT** doppelt an.
 - d. Klicken Sie **OK** an, um das Fenster **SQL Expression Builder** zu schließen. Sie brauchen die Standard-Spaltenberechnungsfunktion **SUM** auf der Seite **Spaltenberechnungen** nicht zu ändern. Die Funktion **SUM** ist die Standardeinstellung für das Maß **Flow Out**, weil die Datenquelle numerisch ist und das Maß auf eine Spalte verweist und nicht nur auf andere vorhandene Maße.

4. Gehen Sie wie folgt vor, um das Maß **Current Value** zu erstellen:
 - a. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnete Maße erstellen** an, um das Maß **Current Value** zu erstellen. Das Fenster **SQL Expression Builder** wird geöffnet.
 - b. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung **CURRENT VALUE** in das Feld **Name** ein.
 - c. Führen Sie die folgenden Schritte aus, um den SQL-Ausdruck 'Current Value' zu erstellen:
 - Erweitern Sie den Ordner **Spalten** und die Faktttabelle in der Liste **Daten**.
 - Klicken Sie das Maß **PRODUCT_VALUE** in der Liste **Daten** doppelt an.
 - Klicken Sie den Operator ***** in der Liste **Operatoren** doppelt an.
 - Klicken Sie die Spalte **CURRENT_QUANTITY** in der Liste **Daten** doppelt an.

Abb. 1 zeigt den Ausdruck 'Current Value', den Sie im Fenster **SQL Expression Builder** erstellen können.

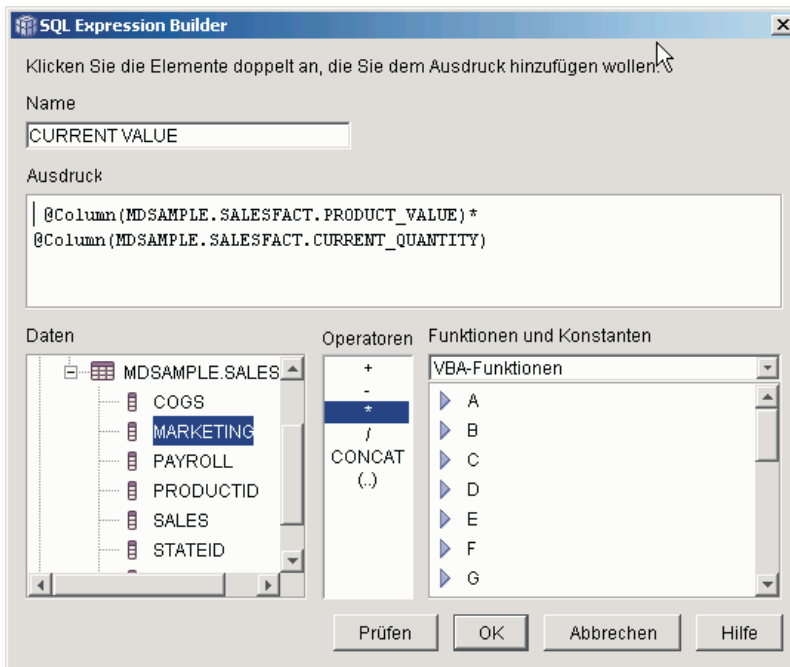


Abbildung 1. Vollständiger Ausdruck 'Current Value' im Fenster **SQL Expression Builder**

- d. Klicken Sie **OK** an, um das Fenster **SQL Expression Builder** zu schließen.
- e. Klicken Sie auf der Seite **Spaltenberechnungen** die Spaltenberechnung für das Maß **Current Value** und anschließend **Spaltenberechnungsprozedur** in der Liste an. Das Fenster **Aggregation Script Builder** wird geöffnet. Die Standard-Spaltenberechnungsprozedur verwendet die Funktion SUM für alle Dimensionen.
- f. Die Zeitdimension muss nach unten versetzt werden. Wählen Sie dazu



Time aus, und klicken Sie den Druckknopf an, um die Zeitdimension an die letzte Position der Dimensionsliste für die Prozedur zu versetzen. Klicken Sie anschließend bei weiterhin ausgewählter Dimension **Time** die Funktion **AVG** in der Liste **Spaltenfunktionen** doppelt an. Die Spaltenberechnungsprozedur (wie in Abb. 2 gezeigt) kumuliert die Daten für alle Dimensionen außer der Dimension **Time**, für deren Daten der Durchschnitt ermittelt wird.

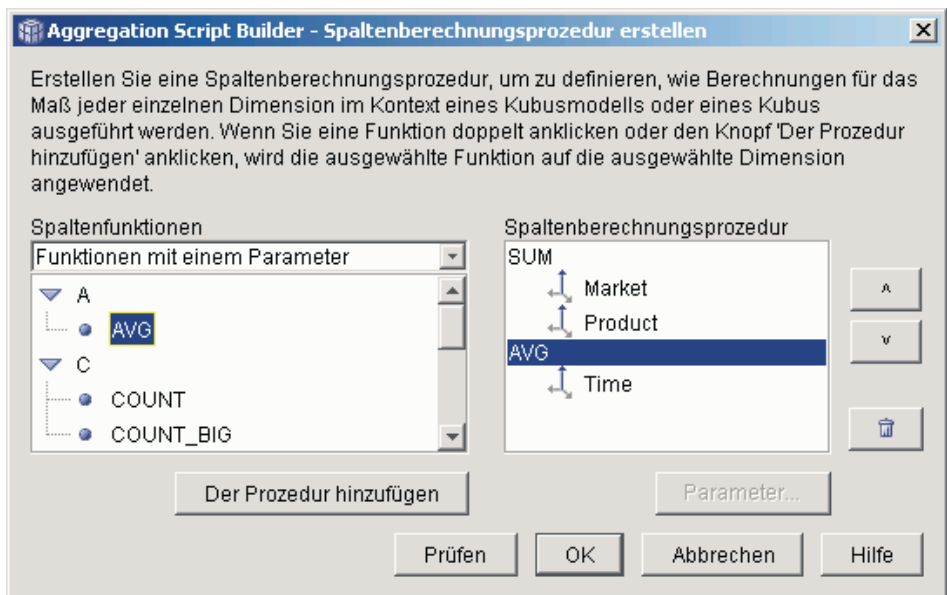


Abbildung 2. Spaltenberechnungsprozedur für das Maß **Current Value**

- g. Klicken Sie im Fenster **Aggregation Script Builder** den Druckknopf **Prüfen** an, um die Gültigkeit der Spaltenberechnungsprozedur zu prüfen. Klicken Sie **OK** an, um die Spaltenberechnungsprozedur zu speichern und das Fenster zu schließen.

5. Klicken Sie **OK** an, um die Änderungen am Faktenobjekt zu speichern und das Fenster **Faktenmerkmale** zu schließen.

Sie verfügen nun über drei berechnete Maße für den Lagerbestand im Warenlager. Mit diesen Maßen können Sie die Fluktuation des Produktbestands in Ihrem Warenlager analysieren.

Kapitel 2. Korrelieren von Werbekosten und Verkaufszahlen

Ein Autohändler hat vor, seine Ausgaben für Werbemaßnahmen zu erhöhen. Um eine fundierte Entscheidung treffen zu können, möchte der Autohändler zunächst die bisherige Entwicklung der Relation zwischen Werbekosten und Verkaufszahlen analysieren. Dabei soll ermittelt werden, ob sich unterschiedlich hohe Ausgaben für die Werbung auf die Verkaufszahlen ausgewirkt haben. Insbesondere soll festgestellt werden, ob höhere Werbeausgaben in direkter Relation zu höheren Verkaufszahlen stehen.

Details des Szenarios

Die Datenbank des Autohändlers verfügt über eine Faktentabelle mit den Spalten **Sales** (Verkauf) und **Ad Costs** (Werbekosten). Die Datenbank enthält außerdem einige weitere Dimensionstabellen. Der DBA kann ein Maß erstellen, das die DB2-Funktion **CORRELATION** verwendet, um die Korrelation zwischen den Kosten und den Verkaufszahlen zu berechnen. Die Funktion **CORRELATION** ist eine Funktion mit mehreren Parametern, die zwei Eingabeparameter benötigt. In diesem Beispiel verwendet der DBA die Spalten **Sales** und **Ad Costs** als die beiden Eingabeparameter.

Der DBA muss die Spaltenberechnungsfunktion mit mehreren Parametern zuerst in der Spaltenberechnungsprozedur anwenden. Die Funktion mit mehreren Parametern kann für alle Dimensionen angewendet werden. Es kann aber auch für die Dimension **Time** eine zweite Funktion angewendet werden (zum Beispiel die Funktion **MAX**) und für alle anderen Dimensionen die Funktion mit mehreren Parametern. Der DBA definiert den SQL-Ausdruck für das Maß, so dass er direkt der Spalte **Ad Costs** zugeordnet wird. Der SQL-Ausdruck ist der erste der beiden Parameter für die Funktion mit mehreren Parametern. Der DBA definiert den zweiten Parameter als einen SQL-Ausdruck, der direkt der Spalte **Sales** zugeordnet wird. Die Funktion **CORRELATION** wird als einzige Spaltenberechnungsfunktion definiert, damit das Maß die statistische Korrelation zwischen den Werbekosten und den Verkaufsergebnissen für alle Dimensionen berechnen kann.

Schritte zum Erstellen des Maßes

Die folgenden Schritte zeigen, wie Sie mit dem Fenster **Faktenmerkmale** der OLAP-Zentrale das Maß **Advertising-Sales Correlation** in einem vorhandenen Faktenobjekt erstellen können:

1. Klicken Sie das Faktenobjekt in der Objektbaumstruktur der OLAP-Zentrale mit der rechten Maustaste an, und klicken Sie **Maße editieren** an, um das Fenster **Faktenmerkmale** zu öffnen.
2. Klicken Sie den Druckknopf **Berechnetes Maß erstellen** an. Das Fenster **SQL Expression Builder** wird geöffnet.
3. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung **ADVERTISING-SALES CORRELATION** in das Feld **Name** ein.
4. Definieren Sie den SQL-Ausdruck für das Maß, der als erster Parameter für die Funktion **CORRELATION** mit mehreren Parametern in der Spaltenberechnungsprozedur verwendet wird. Um den Ausdruck zu definieren, erweitern Sie den Ordner **Maße** in der Liste **Daten**, und klicken Sie das Maß **AD COSTS** doppelt an, um es der Liste **Ausdruck** hinzuzufügen. Abb. 3 zeigt den Ausdruck, den Sie im Fenster **SQL Expression Builder** erstellen.

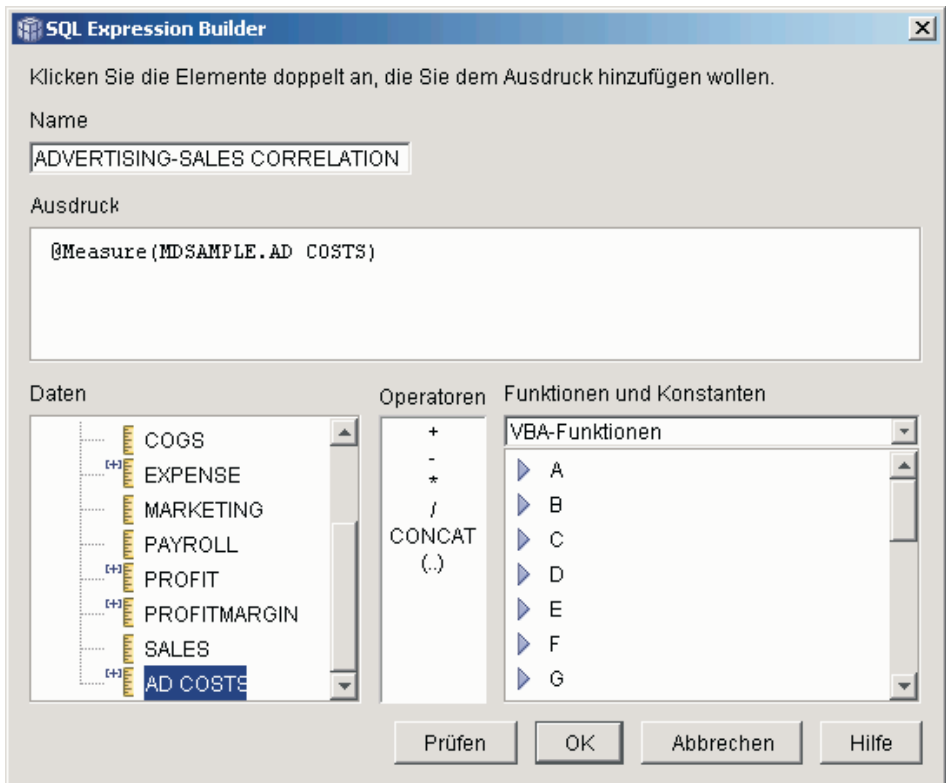


Abbildung 3. Vollständiger Ausdruck 'Advertising-Sales Correlation' im Fenster **SQL Expression Builder**

- Klicken Sie auf der Seite **Spaltenberechnungen** die Spaltenberechnungsfunktion für das Maß **ADVERTISING-SALES CORRELATION** an, und wählen Sie **Spaltenberechnungsprozedur** in der Liste aus, wie in Abb. 4 gezeigt. Das Fenster **Aggregation Script Builder** wird geöffnet.

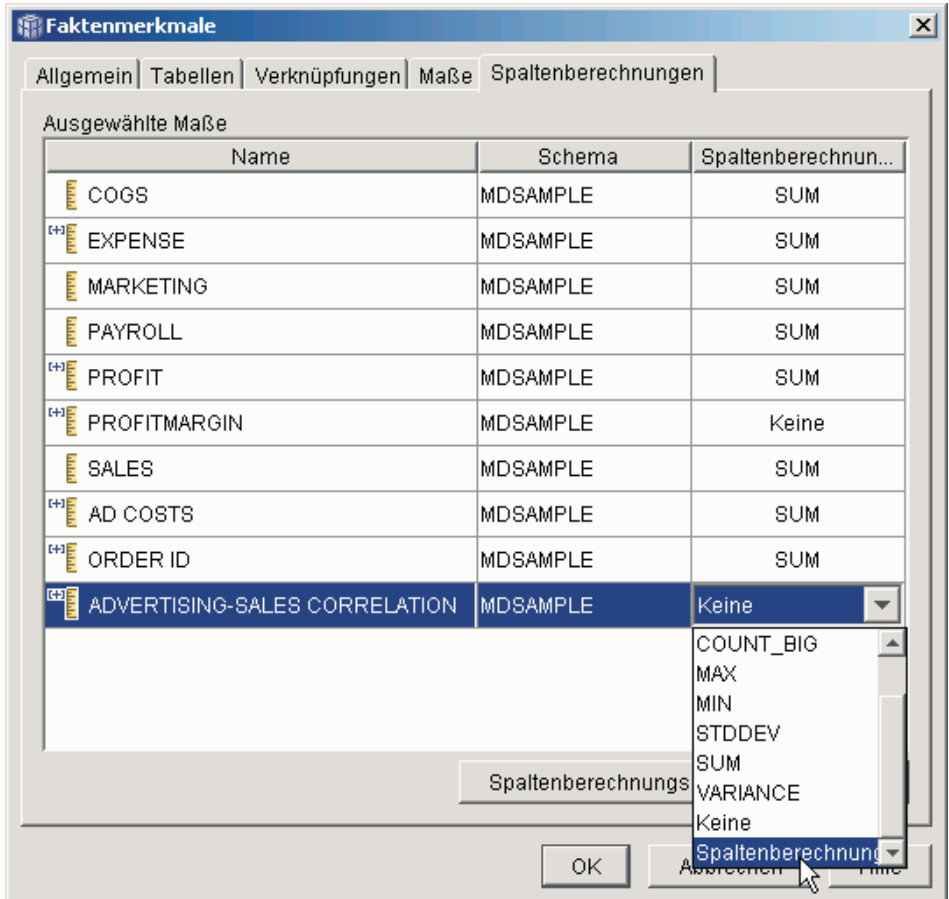


Abbildung 4. Seite **Spaltenberechnungen** des Fensters **Faktenmerkmale**

- Wählen Sie im Feld **Spaltenfunktionen** den Eintrag **Funktionen mit mehreren Parametern** aus. Wählen Sie in der Liste der Funktionen mit mehreren Parametern die Funktion **CORRELATION** aus, und klicken Sie **Der Prozedur hinzufügen** an. Abb. 5 auf Seite 12 zeigt die Funktion **CORRELATION** an oberster Stelle der Dimensionsliste in der Prozedur.

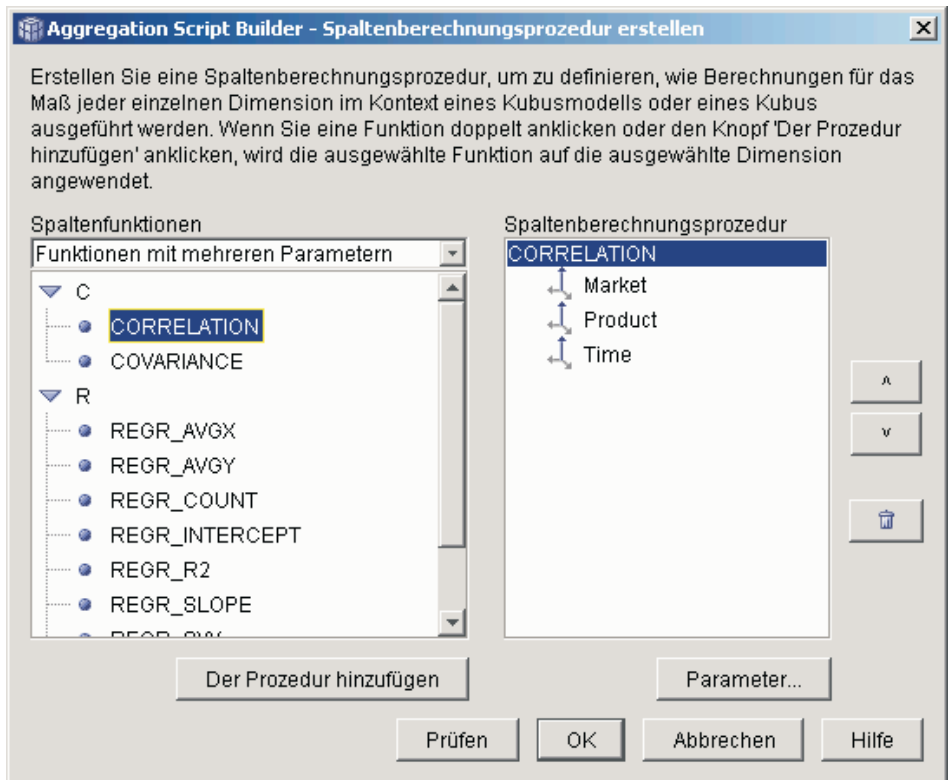


Abbildung 5. Spaltenberechnungsprozedur für das Maß **Advertising-Sales Correlation**

7. Klicken Sie den Druckknopf **Parameter** an, um den zweiten Parameter für die Funktion **CORRELATION** anzugeben. Das Fenster **Funktionsparameter** wird geöffnet. Wählen Sie **Vorhandenes Maß verwenden** und anschließend **SALES** aus. Abb. 6 auf Seite 13 zeigt das Fenster **Funktionsparameter** mit dem als zweiter Parameter angegebenen Maß **Sales**. Klicken Sie **OK** an, um Ihre Auswahl zu speichern und das Fenster **Funktionsparameter** zu schließen.

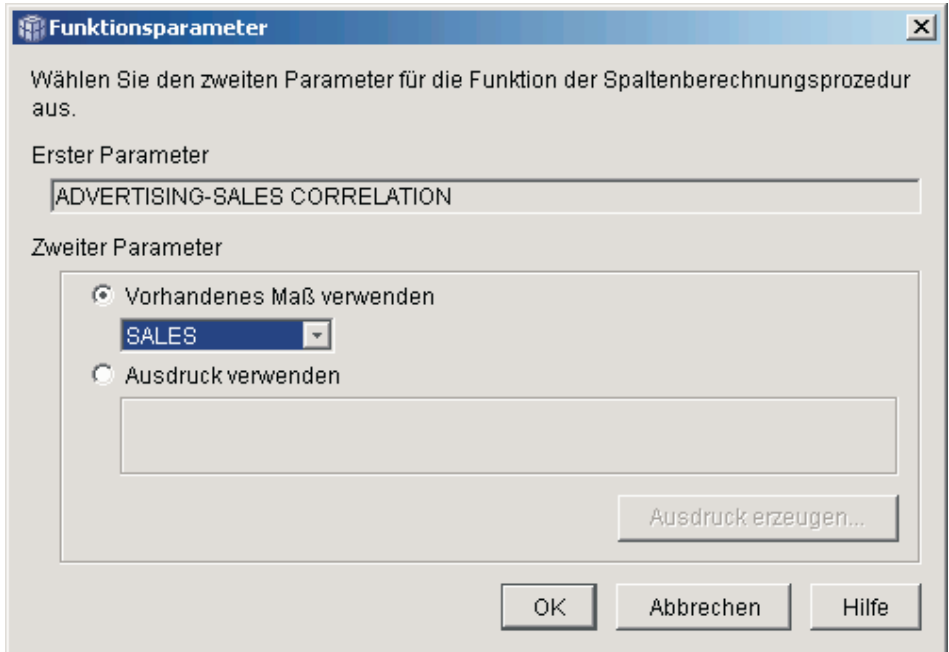


Abbildung 6. Maß **Sales** angegeben als zweiter Parameter im Fenster **Funktionsparameter**

8. Klicken Sie im Fenster **Aggregation Script Builder** den Druckknopf **Prüfen** an, um die Gültigkeit der Spaltenberechnungsprozedur zu prüfen. Klicken Sie **OK** an, um die Spaltenberechnungsprozedur zu speichern und das Fenster zu schließen.
9. Klicken Sie **OK** an, um die Änderungen am Faktenobjekt zu speichern und das Fenster **Faktenmerkmale** zu schließen.

Sie verfügen nun über ein Maß, das zwei Datentypen in Ihrer Datenbank miteinander korreliert. Mit diesem Maß können Sie erfasste Daten auf bisherige Verkaufstrends untersuchen, um Entscheidungen über künftige Werbeausgaben zu treffen.

Kapitel 3. Berechnen des Gewinns und der Gewinnspanne einer Filiale

Der Leiter einer Spielwarenfiliale möchte analysieren, wie sich verschiedene Faktoren, zum Beispiel die Jahreszeit oder der Produkttyp, auf den Gewinn und die Gewinnspanne auswirken. Bevor die detaillierte Analyse durchgeführt werden kann, muss der DBA der Spielwarenfiliale zuerst Maße für Gewinn und Gewinnspanne erstellen. Anschließend kann der DBA weitere Maße erstellen, die verschiedene Faktoren mit den Maßen für Gewinn und Gewinnspanne korrelieren und vergleichen.

Details des Szenarios

Die Datenbank der Spielwarenfiliale verfügt über eine Faktentabelle mit den Spalten **Sales** (Verkauf), **COGS** (Costs of Goods Sold = Kosten der verkauften Waren) und **Expense** (Ausgaben) sowie den zugehörigen Fremdschlüsselspalten für jede der vorhandenen Dimensionstabellen. Der DBA hat bereits die Maße **Sales**, **COGS** und **Expense** erstellt, die den Spalten **Sales**, **COGS** bzw. **Expense** zugeordnet sind. Die Maße **Profit** (Gewinn) und **Profit Margin** (Gewinnspanne) können vollständig aus diesen vorhandenen Maßen erzeugt werden.

Zum Erstellen des Maßes **Profit** erzeugt der DBA ein Maß, das **SALES**-(**COGS**+**EXPENSE**) in dem SQL-Ausdruck berechnet und die berechneten Daten für alle Dimensionen summiert. Das Maß **Profit** kann durch Verweisen auf vorhandene Maße oder Spalten oder auf eine Kombination aus beiden erzeugt werden.

Nach dem Erstellen des Maßes **Profit** kann der DBA das Maß **Profit Margin** erstellen. Das Maß **Profit Margin** ist das als Prozentwert ausgedrückte Verhältnis zwischen zwei vorhandenen Maßen ($\text{Profit} / \text{Sales} \cdot 100$). Es erfordert keine eigene Spaltenberechnungsfunktion. Eine Spaltenberechnungsfunktion ist nicht erforderlich, weil das Maß nur auf andere Maße verweist, deren Daten bereits berechnet sind. Wenn der DBA zum Berechnen eines Verhältnisses ein zusammengesetztes Maß verwendet (ein Maß, das nur auf andere Maße verweist), braucht er keine zusätzliche Spaltenberechnung zu definieren. Die meisten Spaltenberechnungsfunktionen, zum Beispiel **SUM**, bringen für die Berechnung von Verhältnissen keinen Nutzen. Wenn die Spielwarenfiliale beispielsweise in vier aufeinanderfolgenden Quartalen Gewinnspannen von 40%, 32%, 28% und 37% verzeichnet, ergibt sich beim Summieren der Verhältnisse im Zeitverlauf eine Gewinnspanne von 137% für das ganze Jahr, was keinen Sinn ergibt.

Schritte zum Erstellen der Maße

Die folgenden Schritte zeigen, wie Sie mit dem Fenster **Faktenmerkmale** der OLAP-Zentrale die Maße **Profit** und **Profit Margin** in einem vorhandenen Faktenobjekt erstellen können:

1. Klicken Sie das Faktenobjekt in der Objektbaumstruktur der OLAP-Zentrale mit der rechten Maustaste an, und klicken Sie **Maße editieren** an. Das Fenster **Faktenmerkmale** wird geöffnet.
2. Gehen Sie wie folgt vor, um das Maß **Profit** zu erstellen:
 - a. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnetes Maß erstellen** an. Das Fenster **SQL Expression Builder** wird geöffnet.
 - b. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung PROFIT in das Feld **Name** ein.
 - c. Um den Ausdruck 'Profit' zu erstellen, erweitern Sie den Ordner **Maße** in der Liste **Daten**, und führen Sie die folgenden Schritte aus:
 - Klicken Sie das Maß **SALES** in der Liste **Daten** doppelt an, um es dem Ausdruck hinzuzufügen.
 - Klicken Sie den Operator **-** in der Liste **Operatoren** doppelt an.
 - Klicken Sie das Maß **COGS** in der Liste **Daten** doppelt an.
 - Klicken Sie den Operator **+** in der Liste **Operatoren** doppelt an.
 - Klicken Sie das Maß **EXPENSE** in der Liste **Daten** doppelt an.
 - Heben Sie im Feld **Ausdruck** den Teil `@Measure(MDSAMPLE.COGS)+@Measure(MDSAMPLE.EXPENSE)` des Ausdrucks hervor, und klicken Sie den Operator **(. .)** in der Liste **Operatoren** doppelt an, um den ausgewählten Teil des Ausdrucks in runde Klammern zu setzen.

Abb. 7 auf Seite 17 zeigt den Ausdruck 'Profit', den Sie im Fenster **SQL Expression Builder** erstellen können.

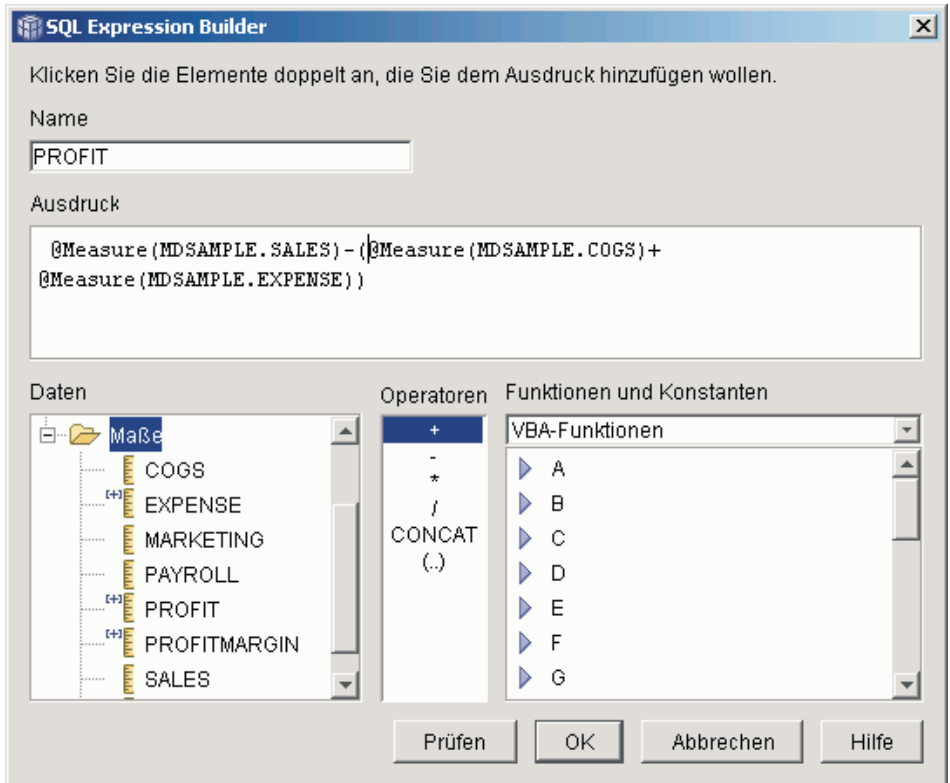


Abbildung 7. Vollständiger Ausdruck 'Profit' im Fenster **SQL Expression Builder**

- d. Klicken Sie **OK** an, um das Maß **Profit** zu erstellen und das Fenster **SQL Expression Builder** zu schließen.
 - e. Klicken Sie auf der Seite **Spaltenberechnungen** die Spaltenberechnung für das Maß **Profit** an, und wählen Sie die Funktion **SUM** aus. Das Maß **Profit** ist damit vollständig.
3. Gehen Sie wie folgt vor, um das Maß **Profit Margin** zu erstellen:
 - a. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnetes Maß erstellen** an. Das Fenster **SQL Expression Builder** wird geöffnet.
 - b. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung **PROFIT MARGIN** in das Feld **Name** ein.
 - c. Um den Ausdruck 'Profit Margin' zu erstellen, erweitern Sie den Ordner **Maße** in der Liste **Daten**, und führen Sie die folgenden Schritte aus:
 - Klicken Sie das Maß **PROFIT** in der Liste **Daten** doppelt an, um es dem Ausdruck hinzuzufügen.
 - Klicken Sie den Operator **/** in der Liste **Operatoren** doppelt an.

- Klicken Sie das Maß **SALES** in der Liste **Daten** doppelt an.
- Schließen Sie den gesamten Ausdruck durch manuelle Eingabe im Feld **Ausdruck** in runde Klammern ein.
- Positionieren Sie den Cursor am Ende des Ausdrucks, und klicken Sie den Operator ***** in der Liste **Operatoren** doppelt an.
- Geben Sie **100** am Ende des Ausdrucks in das Feld **Ausdruck** ein.

Abb. 8 zeigt den Ausdruck 'Profit Margin', den Sie im Fenster **SQL Expression Builder** erstellen können.

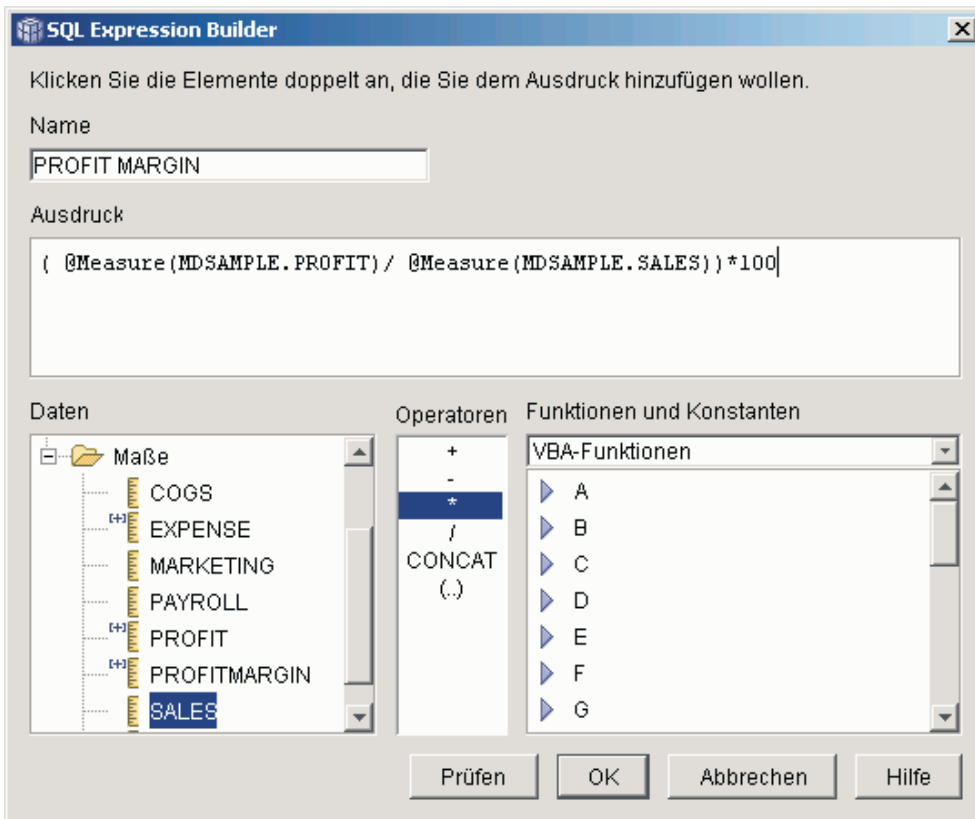


Abbildung 8. Vollständiger Ausdruck 'Profit Margin' im Fenster **SQL Expression Builder**

- d. Klicken Sie **OK** an, um das Maß **Profit Margin** zu erstellen und das Fenster **SQL Expression Builder** zu schließen.

Die Spaltenberechnungsfunktion wird von der OLAP-Zentrale auf der Seite **Spaltenberechnungen** automatisch auf den Wert KEINE für zusammengesetzte Maße gesetzt, d. h. Sie brauchen die Spaltenberechnungsfunktion nicht zu ändern.

4. Klicken Sie **OK** an, um das Fenster **Faktenmerkmale** zu schließen und die beiden neuen Maße zu speichern, die Sie dem Faktenobjekt hinzugefügt haben.

Nachdem der DBA diese beiden Maße erstellt hat, können detaillierte Analysen im Hinblick auf diese beiden wichtigen Maße durchgeführt werden.

Kapitel 4. Zählen der Internetbestellungen

Ein Einzelhandelsunternehmen hat vor ein paar Jahren seinen Geschäftsbe-
reich um Internetverkauf erweitert. Nun möchte das Unternehmen analysie-
ren, wie sich der Internetverkauf entwickelt hat. Dabei ist zunächst zu ermit-
teln, wie viele Bestellungen über das Internet getätigt wurden.

Details des Szenarios

Die Datenbank des Unternehmens verfügt über eine Fakttable für Internet-
bestellungen mit den Spalten `ORDER_ID` (Bestellungs-ID), `PRODUCT_ID`
(Produkt-ID), `QUANTITY` (Menge) und `TIME_ID` (Zeit-ID). Die Spalte `PRO-
DUCT_ID` enthält alle Produkte, die aufgrund einer entsprechenden Bestellung
verkauft wurden, und die Spalte `QUANTITY` erfasst, welche Stückzahl des
Produkts durch die Bestellung erworben wurde. Bestellungen für mehrere
Produkte enthalten eine Zeile für jedes separate Produkt, das in der Bestel-
lung aufgeführt ist. Beispielsweise zeigt Tabelle 6 drei Bestellungen, wobei
Order 1 (Bestellung 1) drei Produkte A, ein Produkt O und ein Produkt G
umfasst.

Tabelle 6. Auszug aus der Fakttable

<code>ORDER_ID</code>	<code>PRODUCT_ID</code>	<code>QUANTITY</code>
1	A	3
1	O	1
1	G	1
2	L	1
2	Q	2
3	P	5

Der DBA kann ein Maß **Order Count** (Bestellanzahl) erstellen, um alle eindeu-
tigen Einträge in der Spalte `ORDER_ID` zu zählen. Das Maß **Order Count**
wird mit dem Schlüsselwort `DISTINCT` im SQL-Ausdruck und der Funktion
`COUNT` für die Spaltenberechnung aller Dimensionen definiert. Der SQL-Aus-
druck für dieses Maß erstellt eine Liste der eindeutigen Bestellwerte, die wäh-
rend der Spaltenberechnung ermittelt werden. Da dieses Maß keine Summen-
funktion beinhaltet, wird es als nicht additives Maß bezeichnet.

Nicht additive Maße sind auch nützlich für Zeichendaten und andere Daten,
die Sie zählen wollen. Beispielsweise können Sie mit nicht additiven Maßen
die Anzahl der Postleitzahlen ermitteln, an die Sie Produkte geschickt haben.

In diesem Beispiel hat der DBA beschlossen, ein Maß **Order ID** zu erstellen, das direkt der Spalte `ORDER_ID` zugeordnet ist. Sie können aber auch die Spalte `ORDER_ID` in dieser Weise verwenden. Die Standardspaltenberechnung ist verschieden, je nachdem, ob in dem SQL-Ausdruck eine Spalte oder ein Maß verwendet wird. In beiden Fällen müssen Sie jedoch die Standardspaltenberechnung auf die Funktion `COUNT` umstellen, wie in „Schritte zum Erstellen des Maßes“ beschrieben.

Schritte zum Erstellen des Maßes

Die folgenden Schritte zeigen, wie Sie mit dem Fenster **Faktenmerkmale** der OLAP-Zentrale das Maß **Order Count** in einem vorhandenen Faktenobjekt erstellen können:

1. Klicken Sie das Faktenobjekt in der Objektbaumstruktur der OLAP-Zentrale mit der rechten Maustaste an, und klicken Sie **Maße editieren** an, um das Fenster **Faktenmerkmale** zu öffnen. Das Fenster **Faktenmerkmale** wird geöffnet.
2. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnetes Maß erstellen** an. Das Fenster **SQL Expression Builder** wird geöffnet.
3. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung `ORDER COUNT` in das Feld **Name** ein.
4. Um den Ausdruck 'Order Count' zu erstellen, erweitern Sie den Ordner **Maße** in der Liste **Daten**, und führen Sie die folgenden Schritte aus:
 - Wählen Sie im Feld **Funktionen und Konstanten** die Option **Verschiedenes** aus. Klicken Sie in der Liste mit verschiedenen Funktionen und Konstanten das Schlüsselwort **DISTINCT** doppelt an.
 - Klicken Sie das Maß **ORDER ID** in der Liste **Daten** doppelt an.

Abb. 9 auf Seite 23 zeigt den Ausdruck 'Order Count', den Sie im Fenster **SQL Expression Builder** erstellen können.

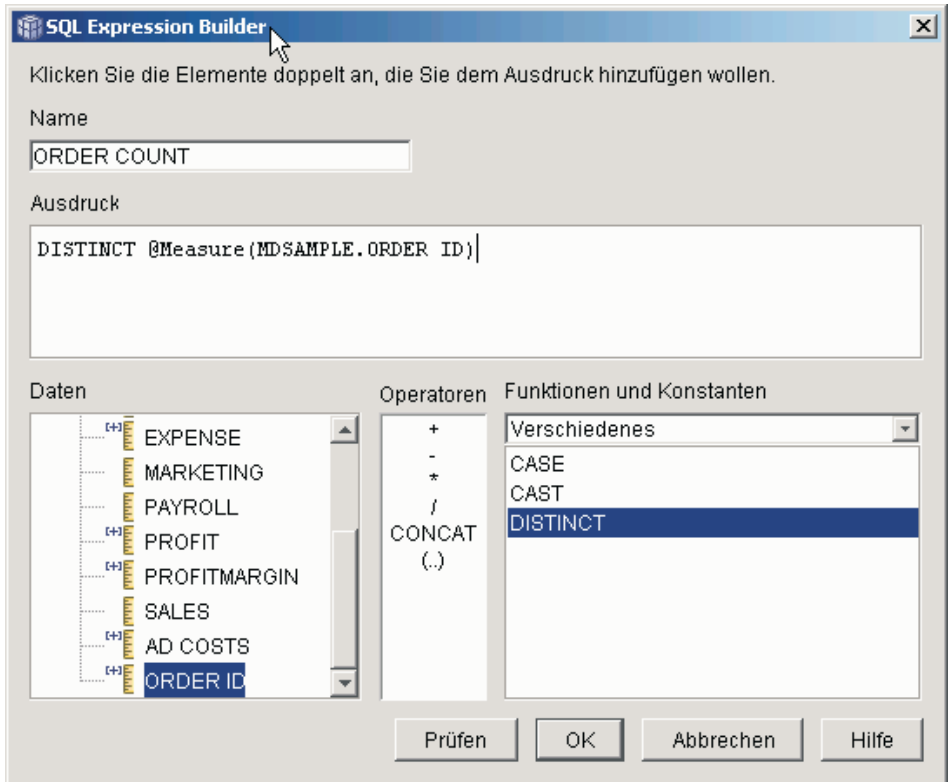


Abbildung 9. Vollständiger Ausdruck 'Order Count' im Fenster **SQL Expression Builder**

5. Klicken Sie **OK** an, um das Fenster **SQL Expression Builder** zu schließen.
6. Klicken Sie auf der Seite **Spaltenberechnungen** die Spaltenberechnung für das Maß **Order Count** an, und wählen Sie die Funktion **COUNT** aus.
7. Klicken Sie **OK** an, um die Änderungen am Faktenobjekt zu speichern und das Fenster **Faktenmerkmale** zu schließen.

Sie verfügen nun über ein Maß zum Zählen der Anzahl eindeutiger Werte in der Spalte **Order ID**. Dieses Maß können Sie zusammen mit anderen Maßen zur detaillierten Analyse Ihrer Daten verwenden.

Kapitel 5. Erstellen einer Rangfolge der Verkaufszahlen

Ein Unternehmen für Bürobedarf hat in den vergangenen Jahren stark expandiert. Die Geschäftsleitung überlegt nun, einige Filialen mit den schlechtesten Absatzzahlen zu schließen, um Kosten einzusparen und den Gewinn zu steigern. Die bisherigen Absatzzahlen der Filialen sind ein wesentlicher Faktor für die Entscheidung, welche Filialen geschlossen werden sollen. Die Analytiker müssen die Möglichkeit erhalten, die Verkaufszahlen in eine Rangfolge zu bringen und detaillierte Analysen für die verschiedenen Dimensionen zu erstellen.

Details des Szenarios

Die Datenbank des Unternehmens für Bürobedarf verfügt über eine Fakt-tabelle mit einer Spalte **Sales** und verschiedenen anderen Spalten. Die Datenbank enthält außerdem mehrere Dimensionstabellen. Der DBA kann ein Maß **Sales Rank** (Verkaufsrangfolge) auf der Basis der Funktion RANK erstellen (eine OLAP-Funktion, die von DB2 Universal Database (DB2 UDB) zur Verfügung gestellt wird).

DB2 Cube Views unterstützt die folgenden OLAP-Funktionen, die von DB2 UDB zur Verfügung gestellt werden:

RANK

Ordnet die Zeilen und weist jeder Zeile eine Rangfolge zu. Die Rangfolge ist definiert als 1 plus die Anzahl vorausgehender Zeilen, die hinsichtlich der Reihenfolge eindeutig sind. Wenn die relative Reihenfolge von mindestens zwei Zeilen nicht ermittelt werden kann, weil sie über doppelte Zeilenwerte verfügen, wird die gleiche Rangfolgenummer zugeordnet. In den Rangfolgeergebnissen können Rangfolgenummern ausgelassen werden, wenn doppelte Zeilenwerte vorkommen. Tabelle 7 auf Seite 27 zeigt die Rangfolgeergebnisse einer Funktion RANK für einige Beispielzeilenwerte.

Die Funktion RANK hat die folgende typische Syntax:

```
RANK ( ) OVER (ORDER BY sortierschlüsselausdruck ausdrucksreihenfolge)
```

Dabei ist *sortierschlüsselausdruck* eine Gruppe von Daten, für die eine Rangfolge gebildet werden soll, und *ausdrucksreihenfolge* ein Schlüsselwort (**ASC** oder **DESC**), das die Werte des Sortierschlüsselausdrucks in aufsteigender oder absteigender Reihenfolge ordnet. Für DB2 Cube Views muss *sortierschlüsselausdruck* ein vorhandenes Maß sein (keine Spalte und kein Attribut). Außerdem unterstützt DB2 Cube Views

nicht die Klausel `PARTITION BY`, die von DB2 UDB mit dieser Funktion zur Verfügung gestellt wird. Weitere Informationen zu der Funktion `RANK` sind in **DB2 UDB Information - Unterstützung** verfügbar.

DENSERANK

Ordnet die Zeilen und weist jeder Zeile eine Rangfolge zu. Die Rangfolge einer Zeile ist definiert als 1 plus die exakte Anzahl der Zeilen, die der Zeile vorausgehen. Die Rangfolgeergebnisse dieser Funktion sind sequenziell und enthalten keine Lücken in den Rangfolgenummern. Tabelle 7 auf Seite 27 zeigt ein Beispiel der Rangfolgeergebnisse der Funktion `DENSERANK` für einige Beispielzeilenwerte.

Die Funktion `DENSERANK` hat die folgende typische Syntax:

```
DENSERANK ( ) OVER (ORDER BY sortierschlüsselausdruck ausdrucksreihenfolge)
```

Dabei ist *sortierschlüsselausdruck* eine Gruppe von Daten, für die eine Rangfolge gebildet werden soll, und *ausdrucksreihenfolge* ein Schlüsselwort (`ASC` oder `DESC`), das die Werte des Sortierschlüsselausdrucks in aufsteigender oder absteigender Reihenfolge ordnet. Für DB2 Cube Views muss *sortierschlüsselausdruck* ein vorhandenes Maß sein (keine Spalte und kein Attribut). Außerdem unterstützt DB2 Cube Views nicht die Klausel `PARTITION BY`, die von DB2 UDB mit dieser Funktion zur Verfügung gestellt wird. Weitere Informationen zu der Funktion `DENSERANK` sind in **DB2 UDB Information - Unterstützung** verfügbar.

ROWNUMBER

Berechnet die sequenzielle Zeilennummer für die Zeile gemäß einer Reihenfolge, die bei der ersten Zeile mit 1 beginnt. Wenn die Klausel `ORDER BY` nicht angegeben ist, werden die Zeilennummern für die Zeilen in zufälliger Reihenfolge vergeben.

Die Funktion `ROWNUMBER` hat die folgende typische Syntax:

```
ROWNUMBER ( ) OVER ([ORDER BY  
sortierschlüsselausdruck ausdrucksreihenfolge])
```

Dabei ist *sortierschlüsselausdruck* eine Gruppe von Daten, für die eine Rangfolge gebildet werden soll, und *ausdrucksreihenfolge* ein Schlüsselwort (`ASC` oder `DESC`), das die Werte des Sortierschlüsselausdrucks in aufsteigender oder absteigender Reihenfolge ordnet. In DB2 Cube Views muss ein vorhandenes Maß (keine Spalte und kein Attribut) als Datenquelle für diese Funktion verwendet werden. Außerdem unterstützt DB2 Cube Views nicht die Klausel `PARTITION BY`, die von DB2 UDB mit dieser Funktion zur Verfügung gestellt wird. Weitere Informationen zu der Funktion `ROWNUMBER` sind in **DB2 UDB Information - Unterstützung** verfügbar.

Diese OLAP-Funktionen sind in der Liste der Funktionen und Konstanten von SQL Expression Builder nicht aufgeführt.

Tabelle 7. Rangfolgeergebnisse für eine Gruppe von Beispielwerten bei Verwendung der Funktionen RANK und DENSERANK

Zeilenwerte	Reihenfolge	Rangfolge- ergebnisse der Funktion RANK	Rangfolge- ergebnisse der Funktion DENSERANK
100	1	1	1
35	2	2	2
23	3	3	3
8	4	4	4
8	4	4	5
6	5	6	6

Schritte zum Erstellen des Maßes

Die folgenden Schritte zeigen, wie Sie mit dem Fenster **Faktenmerkmale** der OLAP-Zentrale das Maß **Sales Rank** in einem vorhandenen Faktenobjekt erstellen können:

1. Klicken Sie das Faktenobjekt in der Objektbaumstruktur der OLAP-Zentrale mit der rechten Maustaste an, und klicken Sie **Maße editieren** an, um das Fenster **Faktenmerkmale** zu öffnen. Das Fenster **Faktenmerkmale** wird geöffnet.
2. Klicken Sie auf der Seite **Maße** den Druckknopf **Berechnetes Maß erstellen** an, um das Maß **Sales Rank** zu erstellen. Das Fenster **SQL Expression Builder** wird geöffnet.
3. Geben Sie im Fenster **SQL Expression Builder** die Bezeichnung SALES RANK in das Feld **Name** ein.
4. Führen Sie die folgenden Schritte aus, um den SQL-Ausdruck 'Sales Rank' zu erstellen:
 - Geben Sie die folgende Funktionssyntax in das Feld **Ausdruck** ein: RANK () OVER (ORDER BY measure DESC).
 - Erweitern Sie den Ordner **Maße** in der Liste **Daten**.
 - Heben Sie das Wort **measure** in der Funktionssyntax im Feld **Ausdruck** hervor, und klicken Sie das Maß **SALES** doppelt an, um es dem Ausdruck hinzuzufügen.

Der vollständige Ausdruck ist in Abb. 10 auf Seite 28 angegeben.

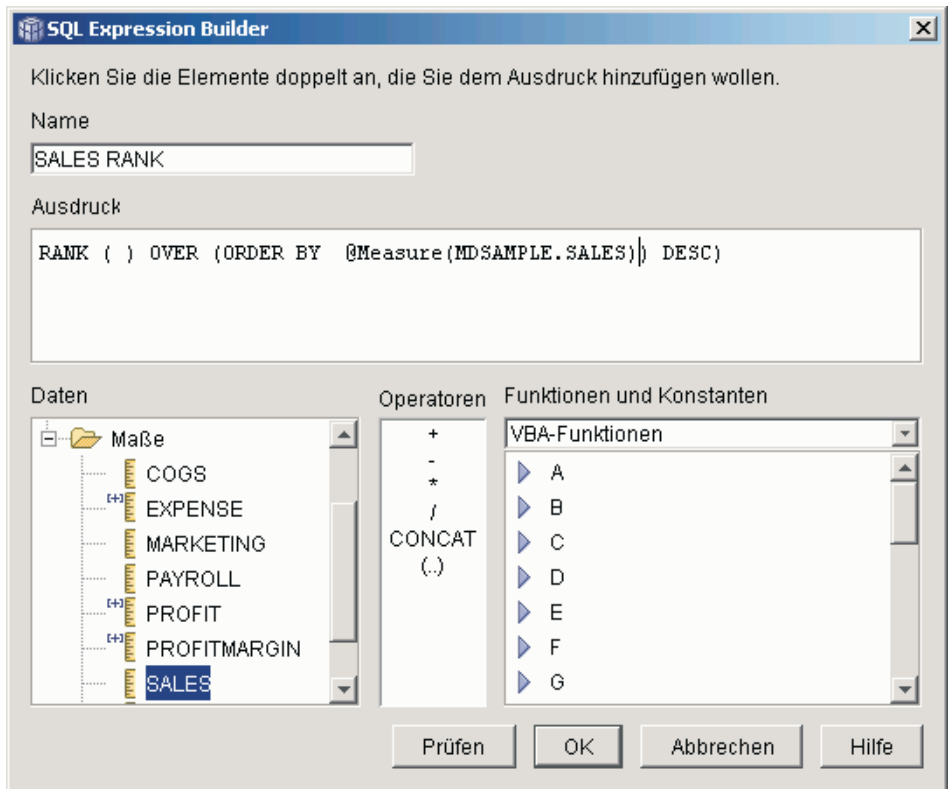


Abbildung 10. Vollständiger Ausdruck 'Sales Rank' im Fenster **SQL Expression Builder**

5. Klicken Sie **Prüfen** an, um sicherzustellen, dass der Ausdruck gültig ist. Klicken Sie **OK** an, um das Fenster **SQL Expression Builder** zu schließen. Sie brauchen die Einstellung 'Keine' für die Standardspaltenberechnung auf der Seite **Spaltenberechnungen** nicht zu ändern. Die Option 'Keine' ist der Standardwert für das Maß **Sales Rank**, weil die Datenquelle numerisch ist und nur auf vorhandene Maße verweist.

Mit der Funktion RANK im SQL-Ausdruck des Maßes können Analytiker die Spalte **Sales** in absteigender Reihenfolge ordnen und unter Einbeziehung der übrigen Dimensionen zuverlässig die Filiale mit den schlechtesten Absatzzahlen für das vergangene Jahr, für eine bestimmte Produktlinie oder für beliebige andere Dimensionsdaten ermitteln, die in der Datenbank gespeichert sind.

Kapitel 6. Erstellen einer Dimension 'Zeit' anhand von Zeitdaten aus der Fakttablelle

Ein Einzelhandelsunternehmen XYZ modelliert die Daten seiner Verkaufstransaktionen mit DB2 Cube Views, um eine effizientere Datenanalyse zu ermöglichen. Da es sich jedoch um transaktionsorientierte Daten handelt, steht als Zeitinformation nur eine Datumsangabe zur Verfügung, die jeder Transaktion zugeordnet ist. In einer Zeitdimension modellierte Zeitdaten werden benötigt, um viele häufig vorkommende Berechnungen in einen genaueren Kontext zu stellen (zum Beispiel die Analyse von Verkaufstrends pro Quartal oder das Berechnen des durchschnittlichen Bestandswerts pro Woche).

Viele DBA speichern Zeitdaten für Transaktionen nicht als Datumsangaben oder Zeitmarken, weil sich Lücken im Datenbestand ergeben, wenn an einem Tag keine Transaktionen stattfinden. Diese Lücken können zu Problemen bei der Spaltenberechnung und dem Anzeigen der Daten führen. In der Regel ist das Modellieren von Zeitdaten in einer Zeittabelle die bessere Lösung. Der DBA des Unternehmens XYZ ist sich aber sicher, dass an jedem Tag mindestens eine Transaktion stattfindet, und er beschließt darum, die aktuelle Datenstruktur beizubehalten.

Details des Szenarios

Das Einzelhandelsunternehmen XYZ verfügt über eine Fakttablelle mit messbaren Daten zu jeder Transaktion, einschließlich 'Sales' (Verkauf), 'Costs' (Kosten), 'Quantity' (Menge) und 'Date' (Datum). Außerdem enthält die Datenbank eine Dimensionstabelle 'Region' (Gebiet) und eine Dimensionstabelle 'Product' (Produkt). Ein Problem besteht darin, dass Zeitdaten nicht in einer separaten Dimensionstabelle gespeichert sind, sondern in der Fakttablelle. Der DBA muss ein Dimensionsobjekt erstellen, das auf den Datumsangaben im Faktenobjekt basiert.

Zum Erstellen einer Zeitdimension, die auf einer einzigen Spalte mit Datumsangaben in der Fakttablelle basiert, müssen zwei Voraussetzungen für die Eindeutigkeit erfüllt sein:

- Da alle Dimensionsobjekte in einem gültigen Kubusmodell mit dem Faktenobjekt verknüpft sein müssen und das Zeitdimensionsobjekt sowie das Faktenobjekt auf derselben Fakttablelle basieren, muss das Zeitdimensionsobjekt durch eine Selbstverknüpfung mit dem Faktenobjekt verknüpft werden, um die Fakttablelle mit sich selbst zu verknüpfen.

- Der DBA muss berechnete Attribute erstellen, die die Datumsangaben zu sinnvollen Einheiten (wie Woche, Monat, Quartal und Jahr) zusammenfassen.

Eine Selbstverknüpfung verknüpft eine Tabelle (in diesem Beispiel die Fakt-tabelle) mit sich selbst. Die Selbstverknüpfung sollte mindestens eine Spalte verknüpfen, die zusammen jede beliebige Zeile in der Fakt-tabelle eindeutig angeben können. Der Primärschlüssel eignet sich hierfür am besten. Wenn kein Primärschlüssel definiert ist, kann als geeigneter Primärschlüsselkandidat diejenige Spaltengruppe gelten, durch die die Fakt-tabelle mit den Dimensionstabelle verknüpft ist. Ein Primärschlüssel muss definiert werden, um das Kubusmodell zu optimieren. Als Verknüpfungskardinalität muss 1:1 und als Verknüpfungstyp eine innere Verknüpfung (Inner Join) verwendet werden.

Abb. 11 auf Seite 31 zeigt, wie ein Faktenobjekt, eine auf der Fakt-tabelle basierende Dimension und eine Fakten-Dimensions-Verknüpfung derselben Fakt-tabelle zugeordnet werden können.

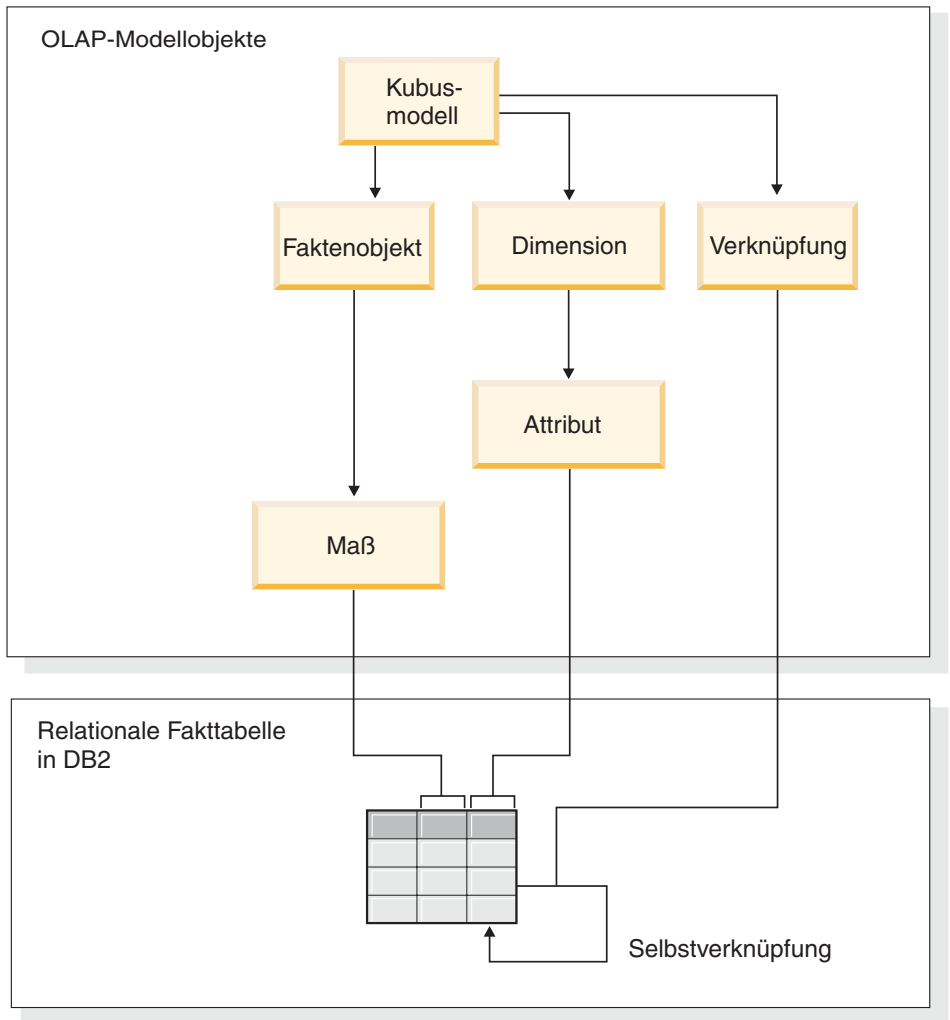


Abbildung 11. Verknüpfung einer Tabelle mit sich selbst durch eine Selbstverknüpfung

Schritte zum Erstellen der Attribute und der Dimension

Die folgenden Schritte zeigen, wie Sie mit dem **Assistenten: Dimension** der OLAP-Zentrale die Dimension **Time** und berechnete Attribute auf der Grundlage der Fakttabelle erstellen können:

1. Klicken Sie zum Öffnen des **Assistenten: Dimension** das Kubusmodell in der Objektbaumstruktur der OLAP-Zentrale mit der rechten Maustaste an, und klicken Sie **Dimension erstellen** an. Der **Assistent: Dimension** wird geöffnet.

2. Geben Sie auf der Seite **Name** die Bezeichnung **Time** in das Feld **Name** ein. Sie können optional den Geschäftsnamen ändern und einen Kommentar eingeben. Klicken Sie **Weiter** an.
3. Wählen Sie die Fakttable des Kubusmodells aus. Klicken Sie **Weiter** an. Sie brauchen keine Dimensionsverknüpfungen anzugeben, weil Ihre Dimension nur eine Tabelle enthält. Klicken Sie auf der Seite **Dimensionsverknüpfungen** den Knopf **Weiter** an.
4. Wählen Sie auf der Seite **Dimensionsattribute** die Spalte für die Zeitmarke aus.
5. Optional: Erstellen Sie weitere berechnete Attribute, um die Zeitmarkendaten zu größeren Einheiten (wie Monat, Quartal und Jahr) zusammenzufassen. Klicken Sie zum Erstellen von berechneten Attributen den Druckknopf **Berechnetes Attribut erstellen** an, um SQL Expression Builder zu öffnen und die SQL-Ausdrücke für jedes Attribut zu definieren, das die Quellenspalte der Zeitmarke in Monate, Quartale und Jahre umrechnet. Klicken Sie nach dem Definieren jedes berechneten Attributs den Druckknopf **Prüfen** an, um sicherzustellen, dass der Ausdruck korrekt ist, und klicken Sie danach **OK** an, um SQL Expression Builder zu beenden und zum **Assistenten: Dimension** zurückzukehren. Klicken Sie nach dem Auswählen und Erstellen aller gewünschten Attribute **Weiter** an.
6. Wählen Sie auf der Seite **Dimensionstyp** die Option **Zeit** aus. Klicken Sie **Weiter** an.
7. Klicken Sie auf der Seite **Fakt-Dimensions-Verknüpfung** die Option **Verknüpfung erstellen** an. Daraufhin wird der **Assistent: Verknüpfung** geöffnet, mit dem Sie die Selbstverknüpfung erstellen können. Geben Sie einen Namen ein, und klicken Sie **Weiter** an. Wählen Sie die Spalte oder Spaltengruppe aus, die jede Zeile in der Fakttable eindeutig definiert (zum Beispiel den Primärschlüssel) sowohl für die linken und als auch für die rechten Attribute aus. Wählen Sie jeweils ein Paar aus, und klicken Sie **Hinzufügen** an, um das Attributpaar der Verknüpfung hinzuzufügen. Wählen Sie als Verknüpfungstyp eine innere Verknüpfung und als Kardinalität 1:1 aus. Klicken Sie nach dem Hinzufügen der erforderlichen Attributpaare **Fertig stellen** an. Der **Assistent: Verknüpfung** wird beendet.
8. Klicken Sie auf der Seite **Fakt-Dimensions-Verknüpfung** den Druckknopf **Fertig stellen** an.

Nach dem Definieren der Dimension **Time** im Kubusmodell kann das Einzelhandelsunternehmen XYZ diese neue Bedeutungsebene in seiner Datenanalyse einsetzen. Nun können zeitbezogene Analysen unter Einbeziehung des Bestands vorgenommen werden.

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Dienstleistungen von IBM verwendet werden können. An Stelle der IBM Produkte, Programme oder Dienstleistungen können auch andere ihnen äquivalente Produkte, Programme oder Dienstleistungen verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Europe
Director of Licensing
92066 Paris La Defense Cedex
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der

ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

Director of Licensing
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer gesteuerten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Marken

Folgende Namen sind in gewissen Ländern (oder Regionen) Marken der International Business Machines Corporation:

DB2
DB2 Universal Database
IBM
Office Connect
Redbooks

Folgende Namen sind Marken oder eingetragene Marken anderer Unternehmen:

Microsoft, Windows, Windows NT, Windows 2000, Windows XP und Microsoft Excel sind in gewissen Ländern Marken oder eingetragene Marken der Microsoft Corporation.

Java und alle Java-basierten Marken und Logos sowie Solaris sind in gewissen Ländern Marken von Sun Microsystems, Inc.

UNIX ist in gewissen Ländern eine eingetragene Marke und wird ausschließlich von der X/Open Company Limited lizenziert.

Linux ist eine eingetragene Marke von Linus Torvalds. Red Hat und alle Red Hat-basierten Marken und Logos sind in gewissen Ländern Marken oder eingetragene Marken von Red Hat, Inc.

Andere Namen von Unternehmen, Produkten oder Dienstleistungen können Marken anderer Unternehmen sein.



Programmnummer: 5724-E15