

**MultiProvider**

MultiProvider sind virtuelle InfoProvider ohne eigene Datenhaltung, deren Hauptaufgabe die Grundlage für die Datenanalyse und -planung ist. Dabei nimmt der MultiProvider zwei wichtige Funktionen wahr: Einerseits dient er als logische Reporting-Schicht und andererseits führt er die Daten unterschiedlicher *InfoProvider* zusammen.

Als logische Schicht nehmen MultiProvider die Rolle einer Schnittstelle zwischen den InfoProvidern mit physischer Datenhaltung und dem Frontend ein. Die Definition einer *BEx Query* oder eines *OLAP-Universums* wird dadurch vom physischen Datenmodell des BW abstrahiert, was mit einer höheren Flexibilität in der Weiterentwicklung des BW-Systems einhergeht. Selbst wenn die Analyse nur auf einem einzigen InfoProvider basiert, ist die Verwendung eines MultiProviders sinnvoll. Selbst bei einer Veränderung oder beim Austausch des zugrunde liegenden InfoProviders ist die Definition der BEx Query oder des OLAP-Universums mit Hilfe des MultiProviders nicht direkt betroffen.

*Logische Reporting-Schicht*

Anwendungen zur Datenanalyse und Planung sollten stets über einen MultiProvider erfolgen. Der Mehraufwand für die Definition und Pflege eines MultiProviders macht sich schnell durch die administrativen Vorteile bezahlt.

**Abb. 9-21**  
Einsatz von MultiProvidern als logische Reporting-Schicht

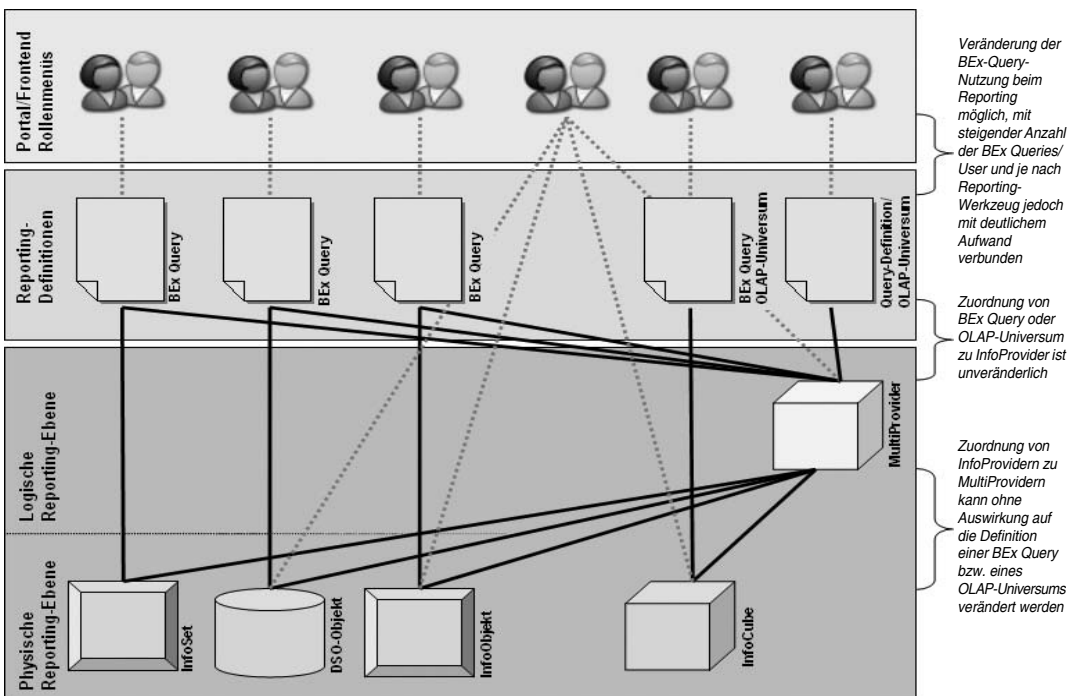


Abbildung 9–21 zeigt den Einsatz eines MultiProviders innerhalb der Datenanalyse, der im Kontext der BW Integrierten Planung hier nicht dargestellte Besonderheiten aufweist. Weitere Details sind in »*Info-Provider in der Planung* → *MultiProvider in der Planung*«, Seite 333 aufgeführt.

Zusammenführung  
von Daten

Die Bezeichnung MultiProvider deutet auf die Hauptaufgabe der MultiProvider hin, die in der Zusammenführung von Daten unterschiedlicher InfoProvider besteht. Die Definition eines MultiProviders kann auf den folgenden InfoProvidern basieren:

- InfoCubes (*Standard-InfoCubes, Real-time InfoCubes, VirtualProvider*)
- InfoSets
- InfoObjekte
- DataStore-Objekte
- Aggregationsebenen der BW Integrierten Planung



Es gibt verschiedene Gründe, Daten in mehreren InfoProvidern zu halten und sie durch einen MultiProvider erst bei der Analyse zusammenzuführen:

1. Die Daten sind aus Designgründen auf mehrere InfoProvider verteilt, z.B. im Falle von Plan- und Istdaten.
2. Es existieren mehrere InfoProvider, die zwar Überschneidungen in den Inhalten aufweisen, deren Struktur aber dennoch heterogen ist. Ein Beispiel hierfür ist die Trennung von Auftragsbelegen in Kopf- und Positionsdaten.
3. Die Daten werden in unterschiedlichen InfoProvider-Typen gehalten, z.B. in Standard-InfoCubes, Real-time InfoCubes, VirtualProvidern und DataStore-Objekten.

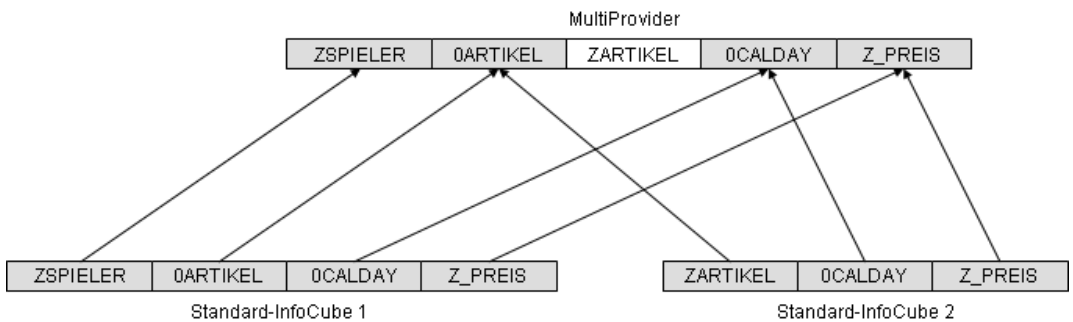
Das Anlegen eines MultiProviders hat große Ähnlichkeit mit dem Vorgehen bei Standard-InfoCubes. In beiden Fällen sind die benötigten *Merkmale* den *Dimensionstabellen* zugeordnet und zusätzlich die *Kennzahlen* und *Navigationsattribute* auszuwählen.

Die Geschwindigkeit einer Anwendung resultiert aus der Lesezeit aus den Datenhaltungsobjekten, wie einem Standard-InfoCube, sowie der Anzahl und Komplexität der nachträglichen Berechnungen in einer BEx Query oder einem Universum. Ein MultiProvider, der auf dem Standard-InfoCube modelliert ist, beeinflusst die Performance nicht<sup>a</sup>. Daher muss die Modellierung nicht nach technischen, sondern kann nach logischen bzw. betriebswirtschaftlichen Kriterien ablaufen.

a. zumindest nicht spürbar

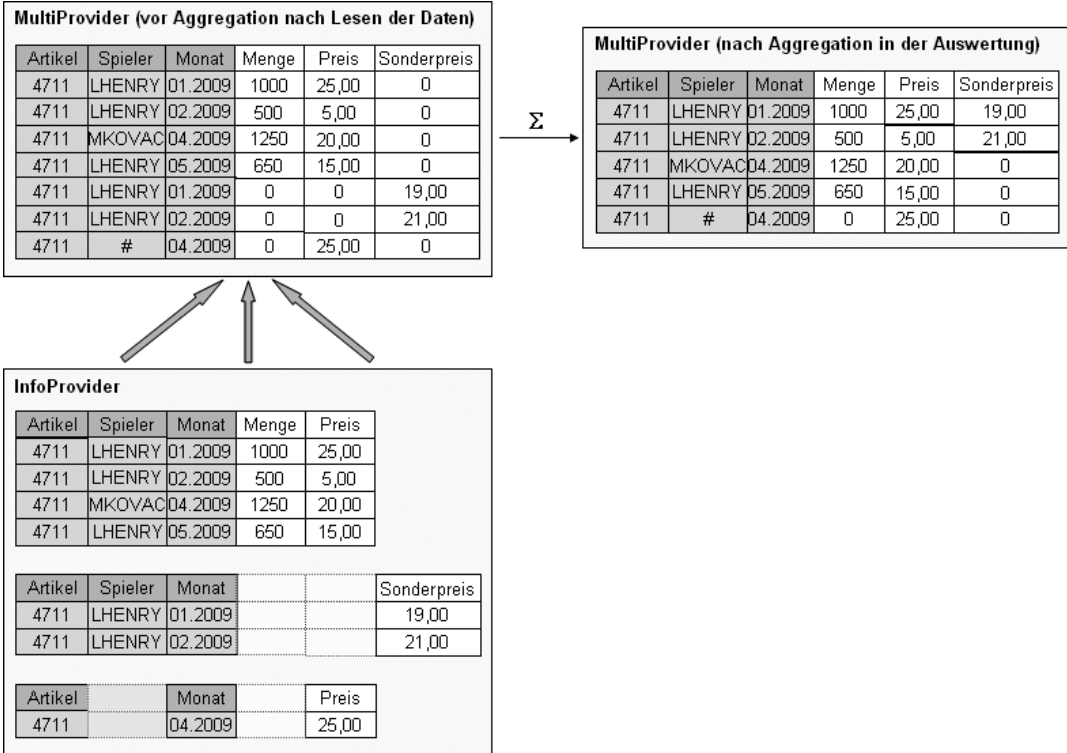
Die Zusammenführung der Daten der zugrunde liegenden InfoProvider wird für jedes InfoObjekt des MultiProviders definiert, wobei jeweils die entsprechenden InfoObjekte der beteiligten InfoProvider (mindestens aus einem) zugeordnet werden. Außer der fachlichen Konsistenz bestehen bei Abbildung 9–22 der InfoObjekte technische Restriktionen, die sich für Merkmale und Kennzahlen unterscheiden: Kennzahlen erlauben die Zuweisung nur von InfoObjekten mit identischem technischen Namen. Merkmale hingegen ermöglichen das Abbilden bereits, wenn InfoObjekte das gleiche Referenzmerkmal (*Referenzierendes Merkmal*) und somit gemeinsame Stammdaten aufweisen.

**Abb. 9–22**  
Identifikation von  
MultiProvider-  
InfoObjekten



Die Zusammenführung der Daten der beteiligten InfoProvider entspricht einer Union-Abfrage. Das heißt, alle InfoObjekte mit identischem technischen Namen (oder gleichem referenzierendem Merkmal) sind aufeinander abgebildet und die Vereinigungsmenge ist aggregiert. Fehlende InfoObjekte in einem InfoProvider werden als initial interpretiert, also mit »#« bei Merkmalen bzw. 0 bei Kennzahlen. MultiProvider (mit Union-Abfrage) stellen als virtuelle InfoProvider quasi das Pendant zu *InfoSets* (mit Join-Abfrage) dar.

</>



**Abb. 9-23** Zusammenführung von Daten durch MultiProvider

## Universum

Universe Designer und Web Intelligence, mittlerweile SAP Business-Objects Web Intelligence genannt, sind die ersten von Business Objects entwickelten Produkte. Ein Universum, das sich mit dem *Universe Designer* erstellen lässt, fungiert als semantische Schicht ohne eigene Datenhaltung, um die Datenquellen für BI-Analyseanwendungen aufzubereiten.

*Universum basierend auf Datenbanken und Dateien*

Universen können viele Datenquellen nutzen, zum Beispiel DB2, Informix, MS Access, MS SQL, MS Excel, Oracle, Teradata, MySQL und andere. Ein Universum stellt die für die Analyse benötigten Tabellen zur Verfügung, indem eine Tabelle angezeigt oder zwei und mehr Tabellen einer Datenquelle mittels SQL-Befehlen miteinander verknüpft werden. Die Befehle sind zum einen manuell erfassbar oder durch die Verbindung bzw. Einstellung der Tabellen generierbar.

*Universum basierend auf OLAP-Systemen*

Mögliche OLAP-Systeme, auf denen ein Universum ebenfalls basieren kann, sind SAP BW, MS Analysis Services 2000, MS Analysis

Services 2005 und Hyperion Essbase. Weitere Informationen sind unter *OLAP-Universum* beschrieben.

**Universum → OLAP-Universum**

Ein OLAP-Universum ist ein *Universum*, das von einem OLAP-Cube oder einer OLAP-Abfrage generiert wurde. Das Universum wird automatisch aus einer ausgewählten Verbindung mit einer OLAP-Datenquelle erstellt. Aus Sicht des SAP BW stehen die nachfolgend aufgelisteten Objekte als Datenlieferanten zur Verfügung:

- *BEx Query*<sup>4</sup>
- *InfoCube*
- *MultiProvider*<sup>5</sup>
- MDX-Statement

Bei der Generierung des BW-Datenlieferanten zu einem Universum werden automatisch die entsprechenden Universumsobjekte zugeordnet. Die Tabelle 9–6 veranschaulicht zusammen mit der Abbildung 9–24 die relevanten BW-Objekte und deren Pendant im Universum.

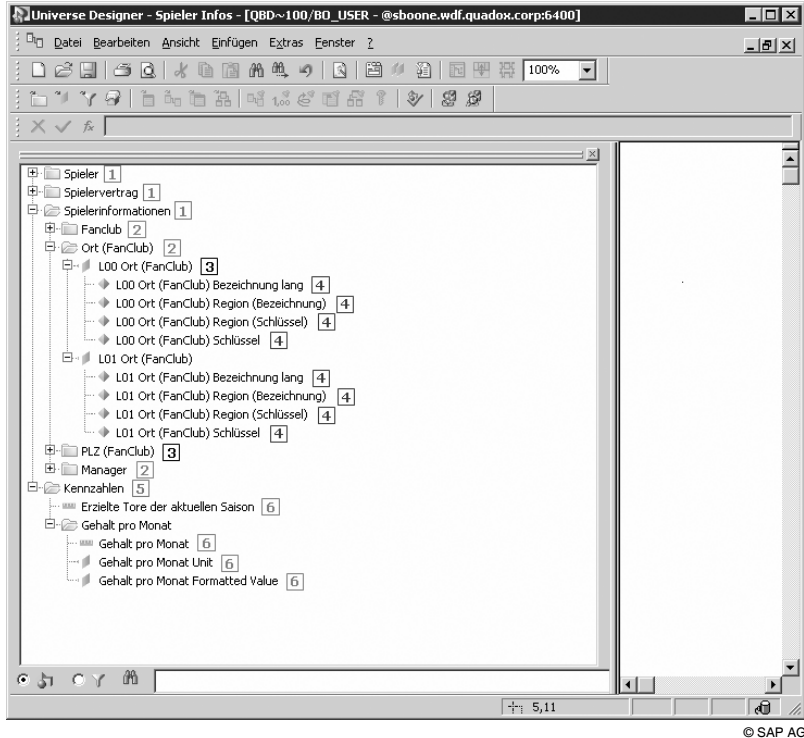
*SAP-BW-Objekte versus Universumsobjekte*

BW-Objekt	Universumsobjekt
<i>Dimension</i> des zugrunde liegenden InfoCubes	Klasse [1]
<i>Merkmal</i>	Unterklasse [2] mit Dimension [3] und deren Detailobjekten [4]
<i>Navigationsattribut</i>	Unterklasse [2] mit Dimension [3] und deren Detailobjekten [4] (wie Merkmal)
<i>Anzeigeattribut</i>	Detailobjekt für die Dimension [4]
Merkmalsstruktur (nur bei BEx Query)	Klasse mit einzeltem Dimensionsobjekt für die Struktur
Kennzahlenstruktur	Klasse [5]
<i>Kennzahl</i>	Kennzahlobjekt in der Klasse für die Kennzahlenstruktur mit Dimensionsobjekten für Einheiten/Währung [6]
Berechnete Kennzahl (nur bei BEx Query)	Kennzahl und Dimensionsobjekte (wie Kennzahl)
Beschränkte Kennzahl (nur bei BEx Query)	Kennzahl und Dimensionsobjekte (wie Kennzahl)

**Tab. 9–6**  
*Relevante BW-Objekte und Universumsobjekte*

4. Um eine BEx Query für das Universum zur Verfügung zu stellen, muss in den *Eigenschaften der BEx Query* für diese BEx Query der externe Zugriff aktiviert sein.
5. Sollte der MultiProvider auf einem *InfoSet* oder einem *DataStore-Objekt* basieren, so kann das OLAP-Universum über diesen Umweg darauf zugreifen.

**Abb. 9-24**  
OLAP-Universum



© SAP AG

OLAP-Universum  
basierend auf BEx Query

Alle InfoObjekte, d.h. Merkmale, Navigationsattribute und Kennzahlen, in der BEx Query, die als Zeilen, Spalten, freie Merkmale und Filter festgelegt sind, stehen im Universum zur Verfügung. Dazu gehören die Eigenschaften und Attribute der Elemente, die Hierarchien, die technischen Namen, die Strukturen und die OLAP-Variablen.

OLAP-Universum  
basierend auf InfoCubes

Für InfoCubes oder MultiProvider werden dem OLAP-Universum bei der Generierung alle Merkmale, Kennzahlen und Hierarchien der Datenquelle zugeordnet.

Verfügbarkeit des  
Universums

Nach dem Anlegen des OLAP-Universums kann es, wie jedes andere Universum, in das Metadaten-Repository des BusinessObjects Enterprise Servers, dem Central Management Server (CMS), exportiert werden. Erst nach diesem Export ist das Universum für die BI-Werkzeuge, zum Beispiel *SAP BusinessObjects Web Intelligence*, zur Erstellung von Analyseabfragen verfügbar.

Anpassen des  
OLAP-Universums

Nach der Generierung des OLAP-Universums lässt sich jede einzelne Komponente modifizieren, indem neue Klassen und Objekte, d.h. die Dimensionen<sup>6</sup>, die Kennzahlen und die Details<sup>7</sup>, hinzugefügt oder bestehende verändert werden. Zu den unterstützten Funktionen

6. Im SAP-BW-Sprachegebrauch auch als Merkmal bezeichnet.

gehören zum Beispiel das Ausblenden, Kopieren und Umbenennen der Universumsobjekte, das Hinzufügen neuer Klassen und Objekte sowie das Bearbeiten von Objektformaten und -Datentypen. Zudem können die Universumsintegrität validiert, Hierarchien bearbeitet, kaskadierende Wertelisten erstellt und die Universumsstruktur regeneriert bzw. aufgefrischt werden.

Wie in Abbildung 9–24 ersichtlich, erstellt der Universe Designer bei der Generierung pro BW-Merkmal zwei Unterklassen bzw. Dimensionen mit entsprechenden Detailobjekten.

- L00<Dimensionsname> →  
Beinhaltet die aggregierte Sicht der entsprechenden Dimension, zum Beispiel für das Merkmal Ort (FanClub) »Alle Orte«.
- L01<Dimensionsname> →  
Enthält alle Werte der entsprechenden Dimension, zum Beispiel für das Merkmal Ort (FanClub) »HD« für »Heidelberg«, »HH« für »Hamburg«, »B« für »Berlin« etc.

Beide Unterklassen bzw. Dimensionen stehen BI-Anwendungen getrennt voneinander zur Verfügung. Allerdings benötigt man die aggregierte Sicht in den seltensten Fällen innerhalb der Ergebnistabelle.

Da sich die aggregierte Sicht auch durch die Summenbildung nachträglich ermitteln und anzeigen lässt und der Endanwender die Namen der generierten Dimensionen in seinen Auswertungen benutzt, ist es empfohlen, die L00-Dimensionen und deren Detailobjekte komplett zu löschen und das Suffix »L01« aus der zweiten Dimension und den zugrunde liegenden Detailobjekten zu entfernen (siehe Abb. 9–25).

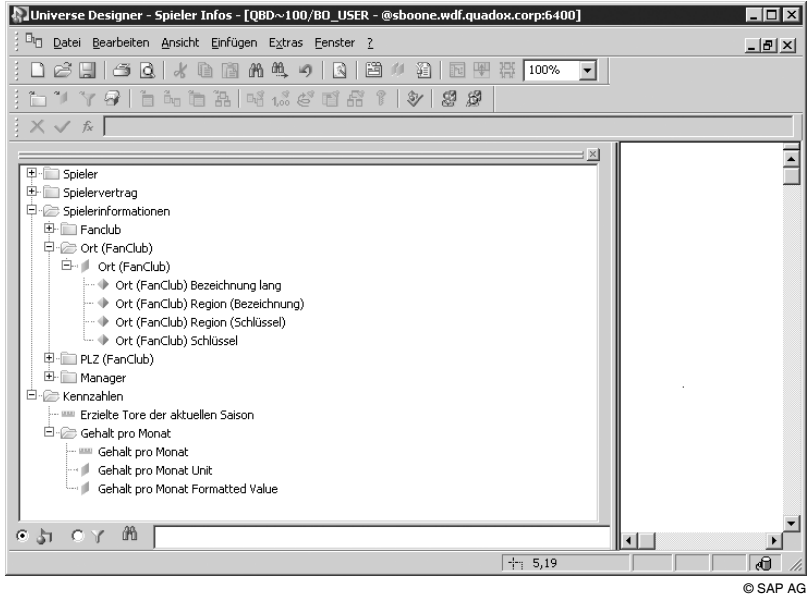
Zusätzlich besteht die Möglichkeit, die technische Komplexität des zugrunde liegenden BW-Datenlieferanten vor dem Endanwender zu verbergen. Hierfür kann der Entwickler die Dimensionen, Kennzahlen und Details innerhalb der Klassen und Unterklassen, ähnlich wie bei einem MultiProvider, nach fachlichen oder betriebswirtschaftlichen Gesichtspunkten umstrukturieren.

Neue Kennzahlen lassen sich, ähnlich wie in der BEx Query, berechnen. Diese berechneten Kennzahlen in OLAP-Universen entsprechen dem Kennzahlobjekt in Nicht-OLAP-Universen mit dem Unterschied, dass die Einschränkung mit Hilfe der in XML-Tags eingeschlossenen MDX-Funktionen statt mit SQL definiert werden.

*Berechnete Kennzahlen  
im OLAP-Universum*

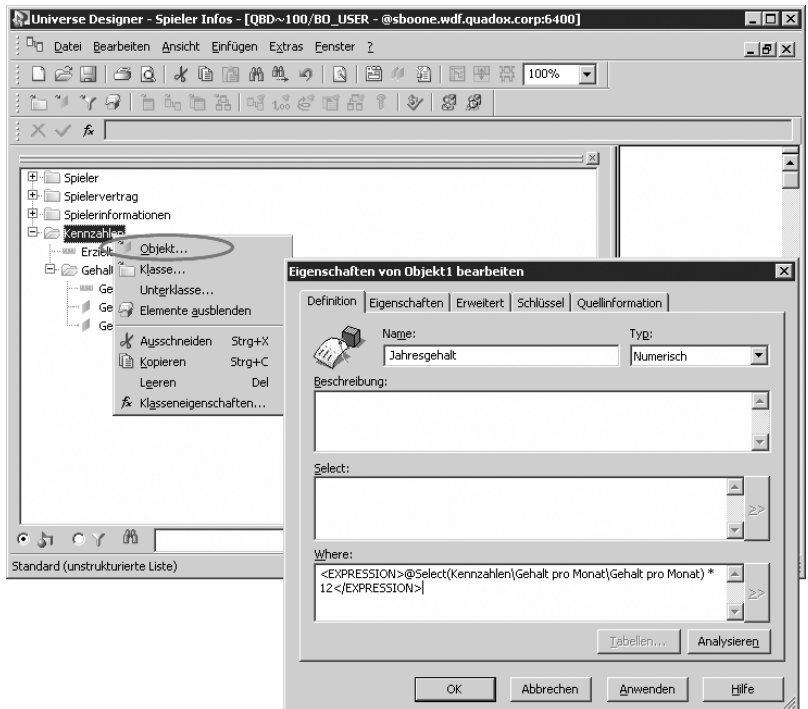
7. Im SAP-BW-Sprachgebrauch handelt es sich um Anzeigeattribute. Navigationsattribute werden, wie in Tabelle 9–6 ersichtlich, wie Merkmale bezeichnet und behandelt.

**Abb. 9-25**  
OLAP-Universum ohne  
aggregierte Werte



© SAP AG

**Abb. 9-26**  
Neue berechnete  
Kennzahl eines  
OLAP-Universums



© SAP AG

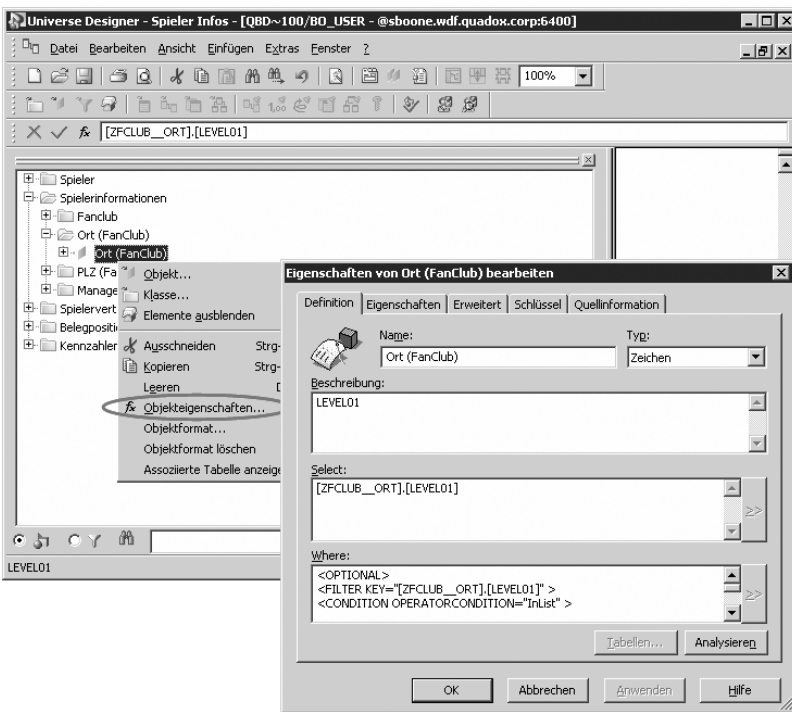
Berechnete Kennzahlen, wie in Abbildung 9-26 dargestellt, werden in Filtern oder der WHERE-Bedingung definiert. Die Syntax für eine berechnete Kennzahl besteht aus Berechnungen, die in die Tags `<EXPRESSION>` `</EXPRESSION>` eingeschlossen sind.

Berechnete Kennzahlen können auf alle OLAP-Metadaten referenzieren, d.h. Dimensionen, Kennzahlen, Details und MDX-Ausdrücke. Konstanten sind im Ausdruck ebenfalls zugelassen, z.B. »2009« oder »ABC«. Durch Prüfen der Integrität werden die XML-Syntax und die Designer-Funktionen validiert.

Designer-Funktionen sind in berechneten Kennzahlausdrücken zugelassen, z.B. @Select, @Prompt, @Variable und @Where (siehe Abb. 9–26).

Von OLAP-Datenquellen generierte Objekte unterstützen optionale Eingabeaufforderungen, die im SAP-BW-Sprachgebrauch auch als Eingabevariablen oder *OLAP-Variablen* mit Verarbeitungsart manuelle Eingabe bezeichnet werden. Für eine optionale SAP-BW-OLAP-Variable wird ein Filter in der WHERE-Bedingung automatisch im Universum generiert. Dieser lässt sich auch manuell nachtragen. Die Syntax für Aufforderungen zur Eingabe in ein OLAP-Universum ist in den Tags <OPTIONAL> </OPTIONAL> eingeschlossen (siehe Abb. 9–27).

*Eingabeaufforderungen  
im OLAP-Universum*



**Abb. 9–27**  
*Eingabeaufforderung  
eines OLAP-Universums*

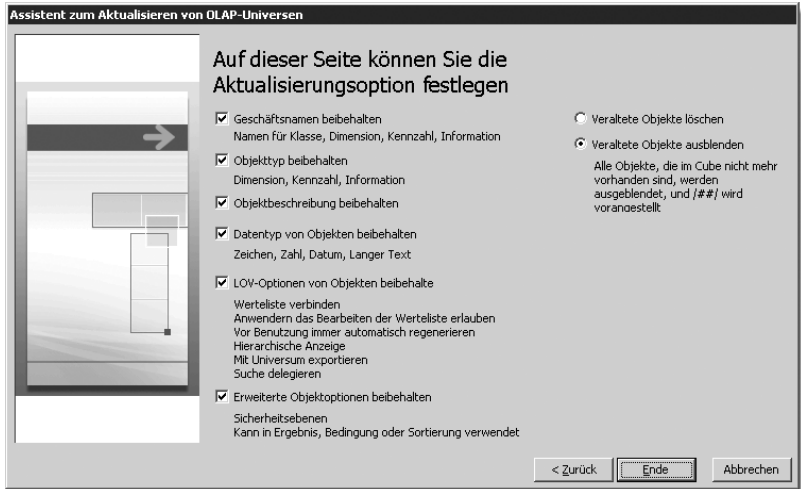
© SAP AG

Mit Hilfe eines Assistenten des Universe Designers lässt sich ein OLAP-Universum automatisch mit den in der Datenquelle vorgenommenen Änderungen aktualisieren. Der Assistent vergleicht das Universum mit dem Datenlieferant und erkennt neue, geänderte und gelöschte Objekte.

*Automatische  
Aktualisierung eines  
OLAP-Universums*

Die dabei einzustellenden Rahmenparameter veranschaulicht Abbildung 9–28.

**Abb. 9–28**  
Einstellungsmöglichkeiten  
zur Aktualisierung eines  
OLAP-Universums



Nach der Generierung des OLAP-Universums können manuell hinzugefügte oder geänderte Objekte beibehalten werden. Veraltete Objekte sind, je nach Wunsch des Entwicklers, aus dem aktualisierten OLAP-Universum zu löschen oder in einer separaten Klasse auszublenden. Das Löschen der Objekte sollte gut bedacht sein, da es in den *Business-Intelligence-Werkzeugen* keinen Verwendungsnachweis, wie sonst im SAP BW bekannt, gibt.

## VirtualProvider

Zu den *InfoProvidern* ohne eigene physische Datenhaltung zählen neben dem *MultiProvider* und dem *InfoSet* der *VirtualProvider mit BAPI*, der *VirtualProvider mit Funktionsbaustein* und der *VirtualProvider mit Staging-Anschluss*.

### VirtualProvider → VirtualProvider mit BAPI



Bei Quellsystemen von Drittanbietern ist für *Bewegungsdaten* ein Direktzugriff durch *VirtualProvider* mit *BAPI* realisierbar. Der *VirtualProvider* ist direkt an das Quellsystem angeschlossen, d. h., eine *Transformation* wird nicht durchlaufen. Daher sind die Daten vom *BAPI* des Quellsystems in der benötigten Form zu übergeben. Die Definition erfolgt als *VirtualProvider* vom Typ »basierend auf einem *BAPI*«.