

DeltaMaster clicks!

07/2014

Liebe Datenanalysten,

ein Ausdruck des unternehmerischen Willens sind Pläne. Wo kommen wir her, wo wollen wir hin? Wir können die Zukunft nicht beliebig gestalten, aber ohne Planung überlassen wir sie dem Zufall – und unseren Wettbewerbern. Pläne geben dem eigenen Handeln eine Orientierung, sie setzen Vergleichsmaßstäbe und oft genug benötigt man sie, um Budgets oder Investoren zu gewinnen. In *DeltaMaster* ist die Planung eng verzahnt mit Analyse und Reporting. Und wie Sie es von diesen beiden Aufgabenbereichen kennen, ist *DeltaMaster* auch in der Planung darauf ausgelegt, möglichst viel Routine an den Rechner zu delegieren. Die Instrumente dafür sind nicht so augenfällig wie manche Analysefunktionen oder Berichtsformate. Aber genau das macht ihren Wert aus: dass sie den Anwendern die Arbeit unaufdringlich erleichtern, damit diese ihre kostbare Aufmerksamkeit ganz ihrer Materie widmen können, nämlich der Planung und nicht der Systembedienung. In diesen *clicks!* stellen wir ein solches Werkzeug vor: die sogenannte Eingabeweiterleitung. Damit lässt sich raffiniert steuern, wie eingegebene Werte ihren Weg in die Datenbank finden. Einmal eingerichtet, geht alles automatisch – deshalb lesen Sie aufmerksam, denn was „unter der Haube“ passiert, werden Sie sonst kaum bemerken.

Herzliche Grüße

Ihr Team von Bissantz & Company



ABC der Management-Information

Auf was muss man achten in der Management-Information? Welche Themen bewegen die Szene? Welche Rolle spielen Auflösung, Blickfolgen oder Skalierung für die Güte von Berichten? Warum spricht man von Ampeln, Tachos oder Dashboards? Ist PowerPoint wirklich gefährlich? All das bewegt uns seit langem und regelmäßig. Im Blog versuchen wir uns an einer kompakten Zusammenfassung: blog.bissantz.de/management-information-1

DeltaMaster-Forum

16. Juli 2014, Neckarsulm

Was ist zu tun, damit Berichte Wirkung entfalten? Diese Frage diskutieren wir mit dem Hirnforscher Professor Dr. Dr. Gerhard Roth und mit den *DeltaMaster*-Kunden Norma Group und WENCO, die ihre Anwendungen und Lösungen vorstellen. Ein weiteres Forum findet am 9. Oktober in Hamburg statt. www.bissantz.de/DeltaMaster-Forum

DeltaMaster-Warm-up

Unser ganztägiges Seminar zum Kennenlernen von *DeltaMaster* bieten wir dieses Jahr noch fünf Mal an: am 18. September in Zürich, am 30. September in Darmstadt, am 16. Oktober in München, am 4. November in Köln und am 5. November in Berlin. www.bissantz.de/dmwu

DeltaMaster@Work

23. Juli 2014, Nürnberg für Interessenten und neue *DeltaMaster*-Anwender www.bissantz.de/dm@w

Archiv

Die *DeltaMaster clicks!* sind über die *DeltaMaster*-Hilfe abrufbar und auf unserer Homepage archiviert. www.bissantz.de/clicks

Kniff des Monats Planung mit DeltaMaster – eingegebene Werte weiterleiten

In vielen Unternehmen wird *DeltaMaster* nicht nur zum Reporting und zur Datenanalyse genutzt, sondern auch zur Planung. Dafür hält *DeltaMaster* eine Reihe von Funktionen bereit, die dem Anwender die Eingabe erleichtern und die Modellierung der zugrunde liegenden Datenbanken vereinfachen. Eine dieser Funktionen ist die sogenannte *Eingabeweiterleitung*: Mithilfe von Regeln lässt sich steuern, wie die eingegebenen Zahlen in Analysewerten erfasst und gespeichert werden. Das Einrichten solcher Regeln gehört zur Entwicklung von Planungsanwendungen – die Endanwender im Modus *Viewer* (die Planer also, die ihre Daten einzugeben haben) bekommen davon erfreulich wenig mit: Ihre Eingaben erscheinen, wo sie hingehören. Welche Regeln im Hintergrund am Werk waren, braucht sie nicht zu interessieren. In diesen *clicks!* werfen wir einen Blick hinter die Kulissen. Wir konzentrieren uns dabei auf die multidimensionale Planung mit Microsoft SQL Server Analysis Services.

Eingeben und Speichern ist zweierlei

Ein eingegebener Wert bezieht sich immer auf einen bestimmten Analysewert und auf eine bestimmte Kombination von Dimensionselementen. Dieser Kontext ergibt sich aus der aktuellen *Sicht* sowie den *Achsendefinitionen* der Pivottabelle. Beispielsweise gibt man den Umsatz (Analysewert) für das Jahr 2015 und in der Wertart Plan (beides Dimensionselemente) ein. Bevor *DeltaMaster* die Eingabe in die Datenbank schreibt, können zusätzliche Verarbeitungsschritte ausgeführt werden – die eingegebene Zahl muss nicht unbedingt so gespeichert werden, wie sie erfasst wurde. Die Verarbeitungsschritte zwischen dem Eingeben und dem Speichern nennen wir *Eingabeweiterleitung*. Die folgenden Verarbeitungen sind möglich:

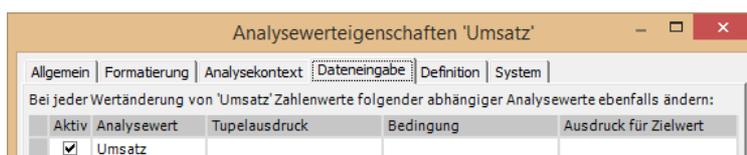
- a) Die Eingabe wird an einen oder mehrere andere Analysewerte weitergegeben und diese werden in die Datenbank geschrieben – anstatt oder zusätzlich zu dem Analysewert, für den ein Wert eingegeben wurde.
- b) Die Eingabe wird an andere Dimensionselemente als die in der *Sicht* und den *Achsendefinitionen* referenzierten weitergegeben.
- c) Die eingegebene Zahl oder die Veränderung gegenüber dem früheren Wert wird umgerechnet, zum Beispiel gerundet.
- d) Die Verarbeitung kann von frei definierbaren Bedingungen abhängig gemacht werden.

Mit Weiterleitungsregeln können also alle drei Parameter der Eingabe beeinflusst werden: der Analysewert, die Dimensionselemente und die eingegebene Zahl. Wir zeigen zunächst, wo und wie Sie Regeln für die *Eingabeweiterleitung* bearbeiten, und gehen dann auf die Einsatzmöglichkeiten ein.

Regeln in den Analysewerteeigenschaften

Die Regeln zur *Eingabeweiterleitung* werden je Analysewert definiert, und zwar für den Analysewert, für den der Anwender einen Wert eingibt, also für den „Ursprung“ einer Weiterleitung. In den *Analysewerteeigenschaften* auf der Registerkarte *Dateneingabe* lassen sich die Regeln in einer Tabelle verwalten (siehe folgende Abbildung; ab *DeltaMaster* 5.6.0). In jeder Zeile steht eine Regel.

Die Spalten haben die folgende Bedeutung:



- Mit dem Kontrollkästchen lässt sich eine Weiterleitungsregel *aktiv* schalten oder temporär außer Kraft setzen, ohne dass die Definitionen verloren gehen.
- Als *Analysewert* ist das Ziel der Weiterleitung einzutragen – ein anderer Analysewert oder derjenige, dessen Eigenschaften man gerade bearbeitet, wenn für diesen bestimmte Dimensionselemente festgelegt oder Bedingungen ausgewertet werden sollen. Dies entspricht Fall a) auf der vorherigen Seite.
- Der optionale *Tupelausdruck* beschreibt, an welche Dimensionselemente eine Eingabe weiterzuleiten ist. Dies entspricht Fall b) auf der vorherigen Seite. Ist das Feld leer, werden Eingaben an die Dimensionselemente weitergegeben, die durch die *Sicht* und die *Achsendefinitionen* der Pivottabelle festgelegt sind.
- Von der optionalen *Bedingung* hängt es ab, ob die Regel ausgeführt und eine Eingabe tatsächlich weitergeleitet wird – Fall d) auf der vorherigen Seite.
- Mit dem optionalen *Ausdruck für den Zielwert* lässt sich der weiterzuleitende Wert individuell berechnen, wie oben als Fall c) erwähnt. Dabei stehen zwei Variablen zur Verfügung: „#old“ für den Wert vor der Dateneingabe, „#new“ für den eingegebenen. Ist das Feld leer, wird eine Wertänderung proportional weitergeleitet.

Einzelheiten sind den *DeltaMaster deltas!* 5.6.0, Punkt 4, zu entnehmen. Die Weiterleitungen können über mehrere Analysewerte verkettet sein: Wenn eine Regel einen Analysewert als Ziel hat, für den ebenfalls Regeln definiert sind, werden diese automatisch mitausgewertet, wobei die berechneten Werte wie eine Benutzereingabe behandelt werden.

Der einfachste Fall: eins zu eins

Im einfachsten Fall wird die Eingabe des Anwenders direkt in die Datenbank geschrieben, „eins zu eins“, ohne weitere Verarbeitung. Dazu sind keine Weiterleitungsregeln erforderlich. Voraussetzung ist, dass der Analysewert und die referenzierten Dimensionselemente direkt in der Datenbank definiert sind (und nicht in *DeltaMaster* angelegt wurden, etwa als benutzerdefinierter Analysewert oder als berechnetes Element). Für einfache Anwendungen, in denen alle Eingaben unmittelbar in die Datenbank übernommen werden können, sind also keine speziellen Vorkehrungen zu treffen.

Eingabe an andere Analysewerte weiterleiten

Etwas anderes ist es, wenn Eingaben auf berechnete Analysewerte ermöglicht werden sollen. Solche Eingaben können nicht direkt in die Datenbank geschrieben werden, da die Analysewerte dort unbekannt sind. Wohl aber ist es möglich, die Eingabe auf Datenbank-Analysewerte „umzuverteilen“, sodass der eingegebene Wert resultiert.

Beispielsweise möchte man im Vertrieb verschiedene Rabattquoten durchspielen. Die Rabattquote wurde in *DeltaMaster* definiert als Quotient aus Rabatt und Umsatz. Diese beiden Analysewerte gibt es auch in der Datenbank, die Rabattquote jedoch nicht. Dennoch können Eingaben für den Quotienten entgegengenommen und verarbeitet werden. In der Abbildung zeigt die hellgraue Schattierung an, welche Felder Eingaben akzeptieren (siehe *DeltaMaster deltas!* 5.5.4, Punkt 11): nur die mit Rabattquoten.

<input type="checkbox"/> Alle Produkte	Umsatz	34.000.000
	Rabatt %	6 %
	Rabatt	2.000.000
<input type="checkbox"/> Luxusmodelle	Umsatz	4.000.000
	Rabatt %	5 %
	Rabatt	200.000
<input type="checkbox"/> Sondermodelle	Umsatz	20.000.000
	Rabatt %	5 %
	Rabatt	1.000.000
<input type="checkbox"/> Standardmodelle	Umsatz	10.000.000
	Rabatt %	8 %
	Rabatt	800.000

Wie sich daraus der Umsatz und der Rabatt ergeben, ist als Regel hinterlegt: Der Umsatz soll konstant gehalten, eine Änderung der Quote allein dem absoluten Rabatt zugeschlagen werden. Die Abbildung zeigt, wie dies zu realisieren ist. Die Regel wird für den Analysewert „Rabatt %“ definiert. Dieser soll nicht selbst geändert werden (daher ist die erste Zeile als nicht aktiv markiert), sondern das Ziel der Regel ist der Analysewert „Rabatt“. Dieser soll berechnet werden als das Produkt aus der „neuen“, vom Anwender eingegebenen Rabattquote und dem konstant gehaltenen Umsatz. Den berechneten Rabatt speichert *DeltaMaster* in der Datenbank und ermittelt damit wiederum die neue Rabattquote, die in der Eingabemaske angezeigt wird. Der Anwender bekommt von dieser Rechenschleife nichts mit: Er gibt eine Quote ein und sieht diese anschließend in seiner Eingabemaske. Dass im Hintergrund der eingegebene Wert auf andere Größen verteilt und diese dann verrechnet wurden, braucht ihn nicht zu kümmern.

Analysewerteigenschaften 'Rabatt %'					
Allgemein	Formatierung	Analysekontext	Dateneingabe	Definition	System
Bei jeder Wertänderung von 'Rabatt %' Zahlenwerte folgender abhängiger Analysew					
Aktiv	Analysewert	Tupelausdruck	Bedingung	Ausdruck für Zielwert	
<input type="checkbox"/>	Rabatt %				
<input checked="" type="checkbox"/>	Rabatt			#new * [Measures].[Umsatz]	
*	<input type="checkbox"/>				

In obigem Beispiel wird also eine Weiterleitungsregel eingesetzt, um Planeingaben für Größen zu ermöglichen, die in der Datenbank nicht existieren. Ein weiterer Anwendungsfall sind abhängige Kennzahlen. Beispielsweise sind Skonto und Rabatt meist eng an den Umsatz gekoppelt. Steigt oder sinkt der Umsatz, so werden Skonto und Rabatt wohl ebenfalls steigen oder sinken. Mithilfe von Weiterleitungsregeln kann *DeltaMaster* diese Anpassung automatisch vornehmen und so dem Anwender die Eingabe erleichtern. Auf einen Ausdruck für den Zielwert kann man hier verzichten – *DeltaMaster* passt die abhängigen Werte dann automatisch proportional an.

Analysewerteigenschaften 'Umsatz'					
Allgemein	Formatierung	Analysekontext	Dateneingabe	Definition	System
Bei jeder Wertänderung von 'Umsatz' Zahlenwerte folgender abhängiger Analyse					
Aktiv	Analysewert	Tupelausdruck	Bedingung	Ausdruck für Zielwert	
<input checked="" type="checkbox"/>	Umsatz				
<input checked="" type="checkbox"/>	Skonto				
<input checked="" type="checkbox"/>	Rabatt				
*	<input type="checkbox"/>				

Die Wirkung dieser Regeln ist in den nebenstehenden Abbildungen zu erkennen: Die Spalte Simulation ist mit den Ist-Werten vorgelegt. Der Umsatz soll um 10 Prozent erhöht werden. Diese Änderung wird automatisch an Rabatt und Skonto weitergeleitet, der Nettoumsatz errechnet sich aus den geänderten Werten.

	Ist	Simulation		Ist	Simulation
Umsatz	8.000.000	10%+	Umsatz	8.000.000	8.800.000
Rabatt	300.000	300.000	Rabatt	300.000	330.000
Skonto	120.000	120.000	Skonto	120.000	132.000
Nettoumsatz	7.580.000	7.580.000	Nettoumsatz	7.580.000	8.338.000

Eingabe mit Bedingungen und variablem Ziel weiterleiten

Zu jedem Speichervorgang gehört eine bestimmte Sicht, wie oben bereits erwähnt, und auch diese kann zwischen Eingabe und Speicherung angepasst werden. Ein typisches Beispiel dafür findet man in der Mehrjahresplanung: Eine häufige Anforderung lautet, dass die nahe Zukunft genauer geplant werden

soll als die ferne. So ist etwa der Absatz für das kommende Jahr nach Monaten aufzuschlüsseln, während für die darauf folgenden Jahre eine Summe für das Gesamtjahr genügt. Solche unterschiedlichen Granularitäten sind in größeren Planungslösungen an der Tagesordnung, haben aber durchaus Tücken, wenn es um die Modellierung in OLAP-Datenbanken geht. Mit Eingabeweiterleitungen lässt sich sehr elegant steuern, wie ein Planwert zu „verbuchen“ ist – zum Beispiel heruntergebrochen auf die einzelnen Monate des nächsten Jahres oder in Summe auf den jeweiligen Dezember der folgenden Jahre.

In diesem Beispiel kommen zwei Optionen der Eingabeweiterleitung gleichzeitig zum Einsatz: zum einen die *Bedingung*, unter der die Regel ausgeführt werden soll, zum anderen der *Tupelausdruck*, der festlegt, für welche Dimensionselemente die Eingabe zu speichern ist. Beide Optionen könnte man auch unabhängig voneinander einsetzen; hier sind sie kombiniert.

Analysewerteigenschaften 'Absatz'					
Allgemein	Formatierung	Analysekontext	Dateneingabe	Definition	System
Bei jeder Wertänderung von 'Absatz' Zahlenwerte folgender abhängiger Analysewerte ebenfalls ändern:					
Aktiv	Analysewert	Tupelausdruck	Bedingung	Ausdruck für	
<input type="checkbox"/>	Absatz				
<input checked="" type="checkbox"/>	Absatz	[Periode].[Periode].CurrentMember	[Periode].[Periode].CurrentMember Is [Periode].[Periode].[Jahr].&[2015]	#new	
<input checked="" type="checkbox"/>	Absatz	[Periode].[Periode].CurrentMember.LastChild.LastChild	Not [Periode].[Periode].CurrentMember Is [Periode].[Periode].[Jahr].&[2015]	#new	

Zwei Regeln benötigen wir für das Beispiel: eine für das nächste Jahr (2015), eine für alle anderen Jahre. Beide Regeln betreffen Eingaben für den Absatz, sind also in dessen *Analysewerteigenschaften* einzutragen, und beide haben den Absatz zum Gegenstand; dieser ist also jeweils als *Analysewert* auszuwählen. Die erste Zeile, ohne *Tupelausdruck* und *Bedingung*, ist deaktiviert, da der Absatz nicht unabhängig von den beteiligten Dimensionselementen geändert werden soll.

Beginnen wir mit der *Bedingung*. Diese ist als MDX-Ausdruck einzugeben, der „true“ oder „false“ zurückliefert. Im Beispiel wird abgefragt, ob sich die Eingabe auf die Periode 2015 bezieht (Zeile 2) bzw. ob sie sich nicht darauf bezieht (Zeile 3).

Im *Tupelausdruck* ist festgelegt, für welche Elemente der eingegebene Wert gespeichert werden soll. Im ersten Fall ist das Jahr 2015 spezifiziert – und dabei wird der Wert gemäß der OLAP-Logik auf die untergeordneten Monate heruntergebrochen („Splashing“). Für alle anderen Jahre wird der eingegebene Wert für Dezember gespeichert (den letzten Monat im letzten Quartal).

Die oben abgebildeten MDX-Ausdrücke sollen das Prinzip verdeutlichen, in der Praxis mag man sie robuster und flexibler formulieren. Auch der *Tupelausdruck* in der zweiten Zeile, „[Periode].[Periode].CurrentMember“ ist nur zur Verdeutlichung eingetragen, denn ein leeres Feld bedeutet ja genau, dass aus jeder Dimension das aktuelle Element zu verwenden ist.

Im Ergebnis sieht man, wie sich eine Eingabe für unterschiedliche Perioden auswirkt: Für 2015 wird die Summe gleichmäßig auf alle Monate verteilt, für 2016 hingegen komplett im Dezember gespeichert.

	2015	Jan 2015	Feb 2015	Mrz 2015	Apr 2015	Mai 2015	Jun 2015	Juli 2015	Aug 2015	Sep 2015	Okt 2015	Nov 2015	Dez 2015	2016	Jan 2016	Feb 2016	Mrz 2016	Apr 2016	Mai 2016	Jun 2016	Juli 2016	Aug 2016	Sep 2016	Okt 2016	Nov 2016	Dez 2016
Absatz	18.000	1.500	1.500	1.500	1.500	1.500	1.500	1.500	1.500	1.500	1.500	1.500	1.500	18.000												18.000

Diese automatische Anpassung der Planungsgranularität lässt sich analog auch für andere Dimensionen realisieren, um zum Beispiel mit Produktgruppen statt Produkten oder mit Regionen statt einzelnen Kunden zu planen.