

Die Qual der Wahl: SAS Grafik oder Power BI Visualisierung per Datenexport?

Thomas RÜDIGER
 AXA Konzern AG
 Thomas.Ruediger@AXA.de

Zusammenfassung

Visualisierungen über Power BI liegen derzeit bei vielen Unternehmen im Trend. Das liegt zum Großteil an der einfachen Bedienungsfläche für Power BI Entwickler, an der Vielfalt von öffentlich zugänglichen Visualisierungs-Apps im Netz, an dynamischen Drill-Down-Bottom-Up-Effekten und der Cloud-Fähigkeit. In diesem Zuge werden Berichte in Excel oder PowerPoint wie auch SAS Berichte abgelöst.

Schlüsselwörter: Microsoft Power BI, Power BI Pro für Consumer, Cloud, SAS Output Delivery System

1 Ausgangstabelle SASHELP.ORSALES

Als Grundlage des Vergleichs von Power BI mit SAS-Grafikprozeduren wird die aus der Tabelle SASHELP.ORSALES erstellte SAS-Tabelle Orion_Sales verwendet.

Sie enthält Jahresumsätze der fiktiven Orion Sales Company.

Im Rahmen einer Datenaufbereitung wurden in einer separaten Tabelle folgende Änderungen gegenüber der Datenquelle durchgeführt:

- Variablen sind deutschsprachig
- Variablen haben deutschsprachige Etiketten (Label)
- Aus Year und Quarter werden zum Testen zwei Variablen mit dem Quartals-Enddatum gebildet mit den SAS-Formaten ddmmyys10. („Power BI-gerecht“) und deudfdd10. („nicht Power BI-gerecht“)
- US-Währungs-Werte sind mit dem Faktor 0,89 auf Euro-Werte umgerechnet.

	Jahr	Quartal	Quartal_Ende	Produkt_Linie	Produkt_Kategorie	Produkt_Gruppe	Anzahl	Gewinn	Verkaufspreis
1	1999	31.03.1999	31/03/1999	Children	Children Sports	A-Team, Kids	286	4.432,33	8.001,90
2	1999	31.03.1999	31/03/1999	Sports	Team Sports	American Football	431	5.331,39	9.441,89
3	1999	31.03.1999	31/03/1999	Outdoors	Outdoors	Anoraks & Parkas	3.139	210.207,16	274.353,17
4	1999	31.03.1999	31/03/1999	Sports	Assorted Sports Arti...	Assorted Sports articl...	4.220	16.761,49	36.761,49
5	1999	31.03.1999	31/03/1999	Outdoors	Outdoors	Backpacks	607	38.323,44	72.328,97
6	1999	31.03.1999	31/03/1999	Sports	Sports	Tennis	125	12.013,89	16.201,38
7	1999	31.03.1999	31/03/1999	Sports	Sports	Soccerball	381	3.208,27	5.735,78
8	1999	31.03.1999	31/03/1999	Sports	Team Sports	Basket Ball	232	2.539,35	4.109,71
9	1999	31.03.1999	31/03/1999	Sports	Swim Sports	Bathing Suits	1.262	23.018,78	39.276,41

Abbildung 1: Arbeitskopie zur Tabelle SASHELP.ORSALES

2 Was wird mit SAS und Power BI verglichen?

Folgende Aufgabestellungen werden exemplarisch herangezogen:

- Allgemeine Auswahlfilter (Datenfilter)
- Jahr (mit Mehrfachauswahl)
- Produktlinie (mit Mehrfachauswahl)

Statistiken

- Verlaufskurve „Gewinn pro Quartal“
- Summentabelle mit den Spalten „Produktlinie/Quartal/Anzahl/Gewinn“
- Kreisdiagramm „Produktlinien“ (Anzahlen)

Zum Vergleich mit Power BI werden häufig genutzte SAS/BASE-Anwendungen verwendet:

- SAS Stored Processes, SAS Web Report Studio, SAS-Macros
- SAS Berichtsprozeduren: PROC PRINT, PROC REPORT, ODS-Features
- SAS Grafikprozeduren, z.B. PROC SGPLOT

3 Berichterstellung mit SAS

In SAS werden Datenfilter und Berichte separat behandelt. Datenfilter lassen sich über Macro-Aufrufe technisch als Macrovariablen und Anwender-freundlich über Stored Processes oder im HTML-Format als web-Seite („_webout“) auswählen. Berichte lassen sich technisch anspruchsvoll über das Output Delivery System („ods“-Statements) erstellen. SAS legt dabei den Fokus auf diverse, mit ODS parallel erstellbare Ausgabeformate (pdf, xls, ...). Interaktive Auswahlboxen inkl. Java-Script-Funktionalitäten (z.B. hide/unhide von HTML-Objekten) sind im web-Bereich (HTML) möglich. Für das Berichtsdesign sind ODS- und Macro-Kenntnisse von Vorteil. SQL und data step mit Daten aus verschiedenen Quellsystemen liefern eine gute Ausgangsbasis für Reporting-systeme. SAS gibt es mit den ersten Berichts- und Grafikprozeduren (proc chart, etc.) seit 1978!

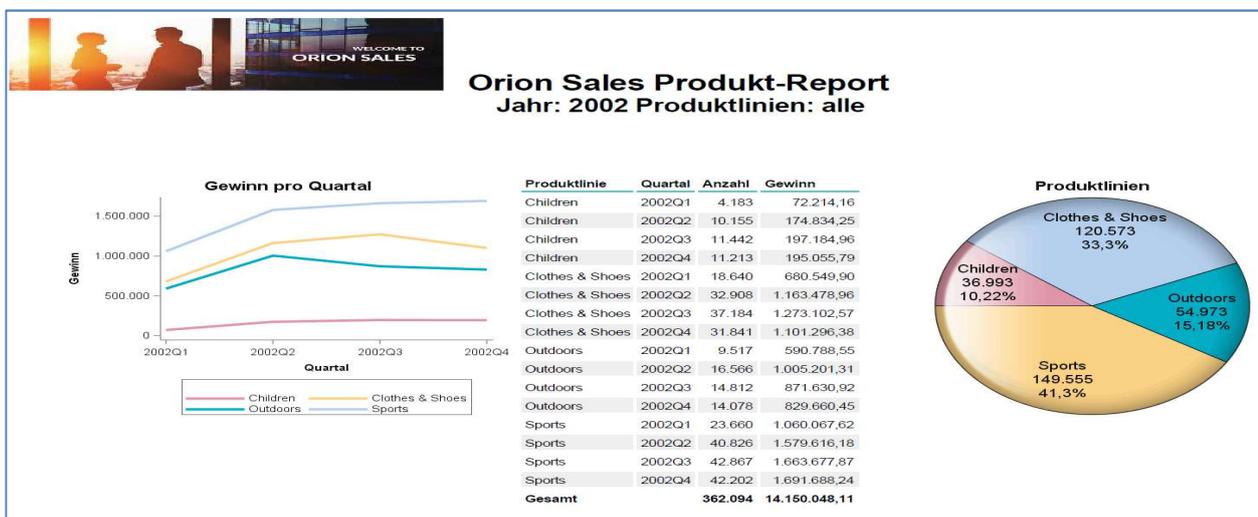


Abbildung 2: SAS-Bericht (pdf-Format)

4 Was ist Power BI?

Power BI ist eine Cloud-basierte analytische Berichts-Plattform der Firma Microsoft zur interaktiven Analyse von Firmendaten, die 2015 zum Abschluss des sog. „Projekts Crescent“ von Ron George ins Leben gerufen wurde. Die zwei wichtigsten Bestandteile sind:

- **Power BI Desktop** – die Windows-basierte Desktop-Anwendung zum Erstellen und Veröffentlichen von Berichten
- **Power BI Pro** für Consumer – Power BI als Cloud-basierte „Software as a Service“

Als Datenquellen können diverse Dateien (Excel, csv, ..) und Datenbanken (Oracle, DB2, ..) dienen.

Der Datenaustausch zwischen Windows-Welt und Cloud gestaltet sich in Form von Arbeitsschritten von Power BI Desktop in Richtung Power BI Pro:

- **1. Power BI Desktop**
 - 1.1 Daten abrufen
 - 1.2 Beziehungen herstellen
 - 1.3 Bericht designen
 - 1.4 Bericht veröffentlichen
- **2. Power BI Pro**
 - 2.1 Arbeitsbereich freigeben (Admin, Mitglied, Mitwirkender, Anzeige)
 - 2.2 App aktualisieren (Navigationsleiste, App-Berechtigungen)
 - 2.3 App verwenden (inkl. Link gegenüber Anwendern kommunizieren)

5 Wie kommen SAS Tabellen nach Power BI?

Zu den Datenimport-Methoden in Power BI gehören neben der Anbindung an Datenbanken wie Oracle oder DB2 u.a.

- Import per R-Skript
- Import einer csv-Datei aus SAS-data step-Export, proc export, ..
- Import eines Excel –Arbeitsblattes (Dateiendung: xlsx)
- Aus einer SAS-Tabelle (analog CSV-Export) Output Delivery System (ods excel), proc export, ..
- Aus einer beliebigen Excel-Datenquelle
- Import über Denodo
 - Denodo ermöglicht Datenintegration ohne Replikation und stellt Daten in Echtzeit bereit, Details, s. <https://www.denodo.com/de>

5.1 Import per R-Skript

Zum Import von SAS-Tabellen muss „R“ auf dem ausführenden Windows Rechner installiert sein. Das Verfahren ist einfach, sicher und ohne SAS-Zusatzexporte (csv, xlsx).

Mit dem nachfolgenden R-Skript wird die Power BI-Tabelle Orion_Sales_R erzeugt. Man beachte, dass in R immer „/“ das Trennzeichen für Verzeichnisse ist.

```
# Das R-Modul sas7bdat muss in R einmalig installiert werden.
# install.packages("sas7bdat")
# rm(list=ls(all=TRUE))
# rm(Orion_Sales_R)
library(sas7bdat)
Orion_Sales_R<-read.sas7bdat("<Quellpfad>/orion_sales.sas7bdat")
```

Nachteile sind:

- SAS-Formate, z.B. numerische Datumsformate oder Benutzer-definierte Formate werden nur mit den Originalwerten ausgegeben, Beispiel: 31.03.1999=SAS-Wert 14334, Lösung: vorheriges Konvertieren in Textvariablen.
- SAS-Begrenzung auf 32 Zeichen bei Variablennamen, Etiketten (Label) können nicht verwendet werden.

5.2 Import einer csv-Datei über einen SAS-data step-Export

Das Verfahren ist über User-Eingabe-freundliche SAS-Macros einfach und sicher. Über die SAS-Funktionen varname, varlabel, vartype und varfmt lässt sich der Datenexport optimieren. Insbesondere können Spaltenetiketten zu Überschriften und numerische Missing-Werte (.,_,.a bis .z) in einfache Leerzeichen übersetzt werden.

Nachteil: Zusätzlicher SAS-Datenexport (Datendoppelhaltung)

```
%macro CSV_Ueberschriften(Was);
  %local RC _I_;
  %if %upcase(&Was.)=NAME %then %let Was=varname;
  %else %let Was=varlabel;
    if _n_=1 then do;
      %let RC=%sysfunc(open(&Ausgabe.));
      %do _I_=1 %to %sysfunc(attrn(&RC.,nvars));
        Zeile=left(trim(Zeile)!!DLM
          !!"%sysfunc(&Was.(&RC,&_I_.))");
      %end;
      %let RC=%sysfunc(close(&RC.));
      Zeile=left(substr(Zeile,2));
      put Zeile;
      Zeile='';
    end;
%mend CSV_Ueberschriften;
```

```
%macro CSV_Daten;
  %local RC _I_;
  %let Was=vartype;
  %let RC=%sysfunc(open(&Ausgabe.));
  %do _I_=1 %to %sysfunc(attrn(&RC.,nvars));
    %if %sysfunc(&Was.(&RC,&_I_.))=N %then %do;
      %if %index(%sysfunc(varfmt(&RC,&_I_.)),COMMA)^=0 %then %do;
        Zeile=left(trim(Zeile)!!DLM
```

```

!!compress (put (%sysfunc (varname (&RC, &_I_.)), %sysfunc (varfmt (&RC, &_I_.)))));
    %end;
    %else %if %index(%sysfunc (varfmt (&RC, &_I_.)), DEUDFDD) ^=0 %then
%do;
    Zeile=left (trim(Zeile)!!DLM
    !!put (%sysfunc (varname (&RC, &_I_.)), ddmmyys10.));
    %end;
    %else %do;
    Zeile=left (trim(Zeile)!!DLM
    !!compress (put (%sysfunc (varname (&RC, &_I_.)), best32.)));
    %end;
    %end;
    %else %do;
    Zeile=left (trim(Zeile)!!DLM
    !!%sysfunc (varname (&RC, &_I_.)));
    %end;
    %end;
%let RC=%sysfunc (close (&RC.));
    Zeile=left (substr (Zeile, 2));
    put Zeile;
%mend CSV_Daten;
%let Ausgabe=WORK.Orion_Sales;
%let Ausgabe_CSV=%sysfunc (pathname (%scan (&Ausgabe., 1, .)));
%let Ausgabe_CSV=&Ausgabe_CSV.\%scan (&Ausgabe., 2, .)_CSV.csv;

data _null_;
    length Zeile $32767;
    retain DLM ";";
    set &Ausgabe.;
    file "&Ausgabe_CSV." lrecl=32767;
    *CSV_Ueberschriften (name);
    %CSV_Ueberschriften (label);
    %CSV_Daten;

run;

```

5.3 Import eines Excel –Arbeitsblattes (Dateiendung: xlsx)

ODS EXCEL in Kombination mit PROC PRINT LABEL ermöglicht den Austausch von Power BI mit Excel. Um fehlende Leerzeichen in den Power BI Spaltennamen zu vermeiden, ist eine Erhöhung der Spaltenbreite über die style-Option width ratsam.

Nachteil: Zusätzlicher SAS-Datenexport, im Unterschied zu csv gelegentliche Power BI-Importfehler.

```

%let Ausgabe_XLSX=%sysfunc (pathname (%scan (&Ausgabe., 1, .)));
%let Sheet_XLSX=%scan (&Ausgabe., 2, .)_XLSX;
%let Ausgabe_XLSX=&Ausgabe_XLSX.\&Sheet_XLSX..xlsx;
%macro Spalte (Name);
    var &Name./style (data)={width=200};
%mend Spalte;
ods results off;ods _all_ close;
ods excel file="&Ausgabe_XLSX.";
ods excel style=dove options (sheet_name="&Sheet_XLSX.");
proc print label data=&Ausgabe.
    noobs width=minimum split="#";

```

```
format Quartal ddmmyy10.;  
%Spalte(Jahr) %Spalte(Quartal)  
%Spalte(Quartal_Ende) %Spalte(Produkt_Linie)  
%Spalte(Produkt_Kategorie) %Spalte(Produkt_Gruppe)  
%Spalte(Anzahl) %Spalte(Gewinn)  
%Spalte(Verkaufspreis)
```

```
run;  
ods _all_ close;ods listing;ods results on;
```

6 Berichterstellung mit Power BI

Vor der Erstellung von Grafiken lassen sich in Power BI eingelesene Tabellen für Auswertungszwecke optimieren.

Power BI Importskripte („Abfragen“) sind mit dem Power Query-Editor hinsichtlich Datenquelle und Feldeigenschaften veränderbar.

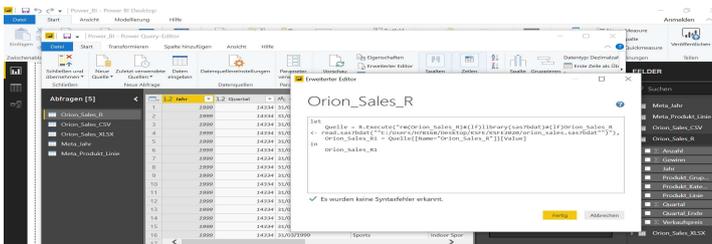


Abbildung 3: Power Query-Editor

Feldeigenschaften (z.B. das Datenformat) sind über den Menüpunkt „Modellierung“ anpassbar.

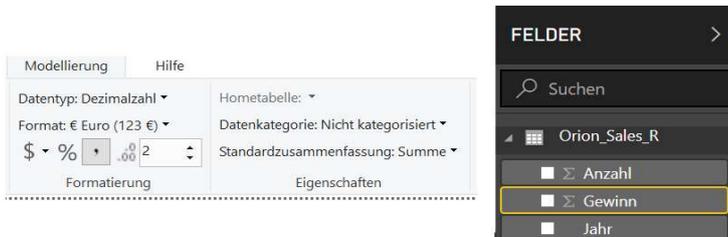


Abbildung 4: Power Bi Modellierung

Tabellenbeziehungen (vor allem n:1-Beziehungen lassen sich über „Beziehungen verwalten“ festlegen.

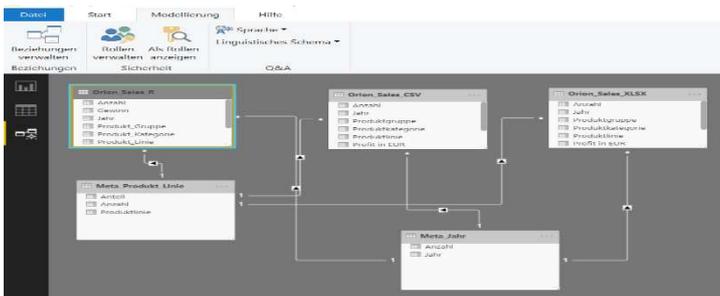
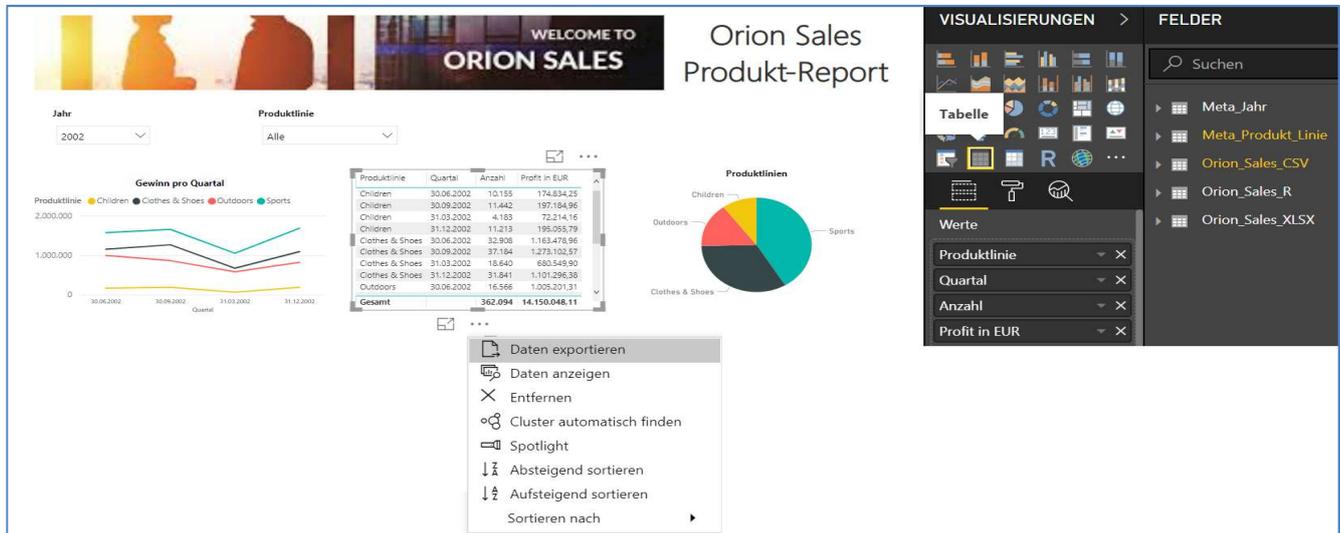


Abbildung 5: Power BI – Beziehungen verwalten

Mithilfe von DAX Funktionen wie divide oder sum lassen sich Anteile und Mittelwerte „on the fly“ berechnen.

Über Visualisierungen lassen sich in Power BI Grafiken und Tabellen mit wenig Aufwand und wenig technischen Skills erstellen.



Drill-Down- und Bottom-Up-Funktionalitäten sind interaktiv über „Datenschnitte“ (Auswahlboxen) und innerhalb von Grafiken und Tabellen möglich.

Tabellen lassen sich mühelos in gängige Formate wie xlsx oder csv und sämtliche Berichte als Powerpoint- oder pdf-Dokument exportieren.

7 Fazit

SAS hat deutliche Vorteile bei der Datenerstellung, Power BI ist dynamischer als SAS (SAS/GRAPH inkl. der SAS-Prozeduren sgplot/ sgrender).

Vor der Erstellung von Grafiken lassen sich in Power BI eingelesene Tabellen für Auswertungszwecke optimieren

Tabelle 1: Vergleich von 7 Aspekten

<i>Aspekt</i>	<i>Power BI</i>	<i>SAS</i>
Datenerstellung/Datenschutz/ Benutzerverwaltung	Kein SQL (Join über Beziehungen), Data Dictionary und Benutzerverwaltung derzeit trotz DAX-Funktion username nicht auslesbar	In SAS gibt es SQL und data step, Data Dictionary entsprechend SASHELP-Views, Benutzerverwaltung über &sysuserid.
Technische Skills fürs Berichtsdesign	Nach anfänglichen Arbeiten (Beziehungen, Modellierung, DAX) niedrig, Aufgabenteilung in Berichtsdesigner und Datentechniker möglich	Hoch (ODS, Macrosprache), Berichterstellung durch Datentechniker. Ggf. sind HTML-Kenntnisse bei Web Reports erforderlich.
Berichtsinteraktivität (Drill Down, Bottom Up)	Dynamisch	Statisch (bei OLAP-Würfeln dynamisch)
Visualisierungsmöglichkeiten und Ausrichtung auf das Corporate Design	Zahlreich mit häufigen, im Web verfügbaren Aktualisierungen	Modular entsprechend Update-Zyklen, fest verdrahtete Images (s. „preimage“-Statement)
Exportfeatures	Nach Powerpoint, PDF Power BI Tabellen separat nach Excel	Parallel erstellbare Ausgabeformate (pdf, xls, ..)
Automatisierbarkeit	Aus SAS-Tabellen hoch, mit Denodo stellt sich die Performance Frage	Hoch über SAS Stored Processes/Macros
Geo-Analysen anhand vorliegender Geo-Koordinaten	Mit Visualisierungen „Zuordnung“/„Flächenkartogramm“ weltweit auf Straßenebene möglich	In Deutschland mit PROC GMAP auf Kreisgemeindeebene, mit Zusatzmodul GIS auf Straßenebene