

Der SAS-Betriebsausflug in die Microsoft-Welt - Datenexport made easy

Thomas Rüdiger
AXA Service AG
Marketing/Direktvertrieb
51171 Köln
thomas.ruediger@axa.de

Zusammenfassung

Für den Wechsel von SAS-Daten in andere Datenwelten (Excel, MSAccess, DBase, CSV, Oracle, DB2, SAP/BW, ...) gibt es vielfältige Möglichkeiten, aber auch Fehlerquellen. Typische User-Fehler sind z.B. abgeschnittene Datenfelder (length Var \$40; format Var \$20.;), unvollständige Exports (>65535 Datensätze, Export nach Excel) oder versehentliches Überschreiben von Dateien. Zudem gibt es sichtbare und unsichtbare Fehlerquellen im SAS-Log: 'Nachname des Kunden' n als gültiger V8-Feldname erscheint sichtbar als Fehlermeldung bei proc export, Felder mit SAS-Datumsformat 'deudfdd' werden mit proc export nicht als Datum übertragen (unsichtbarer Fehler in SAS).

Schlüsselworte: Datenexport, ODBC, DDE, Proc EXPORT, ODS, TSO, XML.

1 Datenwelten sind aufgrund der Libname-Philosophie von SAS leicht erschließbar

Das ‚Look and Feel‘ (z.B. über die SAS-Explorer-Leiste) ist innerhalb der SAS-Umgebung für alle Datenquellen gleich.

Angenehmer Nebeneffekt: Ein Spezialwissen über Flatfile-Struktur in DB2 oder hybride Modellierung in SAP/BW ist über das Libname-Statement für SAS-User redundant.

Eine SAS-Library kann abhängig von Schreib- und Leserechten gleichzeitig für Import und Export genutzt werden.

Der Datenaustausch für Data Reports kennt im wesentlichen nur die Exportrichtung (ODS, data step).

2 Datenexporte

2.1 Export Wizard / PROC EXPORT

Die Libname-Philosophie und Output Delivery System hat sich bei PROC EXPORT nicht durchgesetzt. Über vier hintereinandergeschaltete Frames werden aus SAS heraus je nach Lizenzierung Exportprogramme (proc export) für MSAccess-, Excel-, DBase, Lotus-Dateien oder komma-separierte Textfiles vorbereitet.

Für die gängigen Ausgabeformate läßt sich der vom Export Wizard erzeugte Code als Macro sowie über die Änderung von Input und Output wiederverwenden.

Der Export Wizard benutzt V6-Klassen und V6-Methoden und ist deshalb nicht mehr ‚state of the art‘. Außerdem sind ODBC als Werkzeug oder HTML, PDF, RTF, XML oder TSO als Ausgabeformate für den Export Wizard Fremdworte.

Ebenso ist der Export Wizard nicht ‚customizable‘ (z.B. durch Add-ins oder ODS-Elemente wie title/footnote/style). Die fehlende SAS-Option für Excel-Arbeitsmappen innerhalb Excel-Files erlaubt leider nur einmappige Excel-Files.

PROC-Export-Beispiele:

* Excel;

```
PROC EXPORT DATA=SASHELP.CLASS OUTFILE= "class.xls" <OUTTABLE  
fehlt!!> DBMS=EXCEL REPLACE;  
RUN;
```

* MSAccess;

```
PROC EXPORT DATA=SASHELP.CLASS OUTTABLE= "Sheet"  
      DBMS=ACCESS REPLACE;  
      DATABASE="class.mdb";  
RUN;
```

* DBase;

```
PROC EXPORT DATA=SASHELP.CLASS OUTFILE= "class.dbf"  
      DBMS=DBF REPLACE;  
RUN;
```

* CSV;

```
PROC EXPORT DATA=SASHELP.CLASS OUTFILE= "xxx.csv"  
      DBMS=DLM REPLACE;DELIMITER=';';  
RUN;
```

2.2 SAS/ODBC (Open Database Connectivity)

SAS/ODBC (Open Database Connectivity) verkörpert unter den SAS-Export-Tools die Libname-Philosophie. Als Libname lassen sich Metadaten (Tabellen, Felder) bequem verwalten, die Datenquellen lassen sich aus der Systemsteuerung entnehmen. Data Step- und SQL-Operationen sind direkt beim Lesen und Schreiben von ODBC-Quellen möglich. Ausnahme:

Für Textfiles ist das put- bzw. input-Statement bequemer als ODBC oder PROC EXPORT.

ODBC-Beispiele:

```
%let DBasePath=<Verzeichnis>;
%let File=<Filename>;
* Excel;
libname _ODBC_ odbc complete="dsn=Excel-Dateien;dbq=&FILE.;FIL=Excel 97";
* MSAccess;
libname _ODBC_ odbc complete="dsn=Microsoft Access-Datenbank;dbq=&FILE.";
* DBase;
libname _ODBC_ odbc complete="dsn=dBASE-
Dateien;dbq=&DBasePath."preserve_tab_names=yes;
```

2.3 DDE (Data Dynamic Exchange)

DDE (Data Dynamic Exchange) nutzt die Arbeitsumgebung und Funktionen innerhalb Excel und Word und SAS als externe Datenschleuse.

DDE war Excel vor Excel 7.0 inklusive der heutzutage in Excel versteckten Excel 4.0-Makros vertraut. Der Fortbestand der DDE-Applikationen ist Excel- UND SAS-abhängig und könnte zur technischen Ruine werden.

Pluspunkte hat DDE beim Datenimport (wenn ODBC oder PROC IMPORT erfolglos blieben)

DDE-Beispiel:

```
%let DATA=<Excel-File>;
%let Sheet=<Sheetname>;
filename _DDE_ dde "excel|system";
```

```

data _null_ ;
    file _DDE_ ;
    put '[QUIT()]' ;
run;
x start excel "&DATA.";
options xwait noxsync;
filename _DDE_ dde "excel|&Sheet.!Z1S1:Z1S1";
data _null_ ;
infile _DDE_ ;
input;
stop;
run;
filename _DDE_ clear;

```

2.4 Vergleich PROC EXPORT-ODBC-DDE

In vielen praktischen Anwendungsbereichen erweist sich ODBC gegenüber PROC EXPORT und DDE als vorteilhaft

Tabelle 2.4.1: Vergleich PROC EXPORT-ODBC-DDE

	PROC EXPORT	ODBC	DDE
Excel-Schnittstelle	ja, ohne Export in n>=2 Sheets	ja	ja
MSAccess-Schnittstelle	ja	ja	nein
DBase-Schnittstelle	ja mit OEM/ANSI- Problematik (Umlaute)	ja	nein
data step-Statements/ SQL-Operationen 'on the fly'	nein	ja	ja
freie Feldnamen options validvarname=any;	nein	ja	ja
Umsetzung europäischer V8-Datumsformate wie deudfdd	nein	ja im Datastep	ja im Datastep

3 Textfile-Exporte

3.1 Textfiles/TSO

Neben den Plattform-unabhängigen Textfiles bietet das Filename-Statement mit der ftp-Option und dem put-Statement im data step eine einfache Möglichkeit, Dateien von und auf den Host zu übertragen.

File/TSO-File Beispiel:

```
filename TSO ftp "<Host-File>" host='<Hostname>' user='<UserId>'  
pass='<Password>' lrecl=650 rcmd='site sbdataconn=(iso8859-1,ibm-273)' debug;
```

```
data _null_;  
set SASHELP.Class;  
file TSO;  
put  
'Name'n $char8. /*1-8*/  
'Sex'n $char1. /*9-9*/  
'Age'n 12.2 /*10-21*/  
'Height'n 12.2 /*22-33*/  
'Weight'n 12.2 /*34-45*/  
;run;
```

```
filename TSO clear;
```

3.2 ODS (Output Delivery System)

Für PDF, XML, HTML, ...-Formate orientiert sich ODS (Output Delivery System) an den gängigen Dokumentationssystemen

ODS hat eine Reporting-optimierte SAS/Base-Syntax ohne benutzerfreundliche Oberfläche. In die Ausgabe-Engines (HTML, PDF, ...) können parallel Reports exportiert werden.

ODS-Beispiel:

```
ods html close;
title1 j=c "SASHELP.CLASS";
footnote1 j=c "SASHELP.CLASS";
ods listing close;
ods results off;
ods html file="class.htm"
stylesheet=(url="Styles_Festival.css")
style=Styles.Festival;
ods markup file="class.xml" tagset=Tagsets.Chtml style=Styles.Analysis;
ods proclabel=";
```

```
proc report nowd headskip missing data=SASHELP.CLASS
contents="Report";
run;

title;
footnote;
ods html close;
*ods listing;
ods results on;
```

SAS-Bugfixes und Support sind V9-orientiert,
siehe WARNING bei XML in V8:
WARNING: MARKUP is experimental in this release.

Auch stilistische Elemente wie Inline Text Entry erhalten Einzug in die SAS-Welt
Prosatexte lassen sich mit der escapechar-Option in ods stylen.
Fonts, Schriftgrößen und andere Designelemente lassen sich mit ods escapechar si-
multan für alle Ausgabeformate mit Designer-freundlicher Syntax erzeugen.

ODS Escapechar-Beispiel:

```
ods escapechar="^";
options orientation=portrait;
ods listing close;
ods results off;
ods escapechar='^';
```

```
data Class;
Name='^S={background=red}S^S={font_weight=bold}A^S={font_style=italic}S'
!!byte(32)!!'^S={foreground=blue}S^S={font_width=normal}A^S={font_size=10}S'
;
run;
ods html file="class.html" style=festival;
proc report nowd data=Class;
    column Name;
    define Name/";
run;

ods html close;
```

3.3 Export mit AF

SAS/AF erlaubt das Customizen von Daten- und Reportschnittstellen

um ...

die Vorteile von ODBC,

die Vorteile von SAS-V8/AF gegenüber dem SAS Export Wizard,

ODS und FTP für die Textfile- und Dokumentenverwaltung,

automatisch generierbaren Code,

und die sonstigen Bedürfnisse im Arbeitsumfeld (Usability, Customizability)

... optimal zu erschließen

Ausblick: Ein optimierter AF-Datenimport bietet sich als Folgemaßnahme an