

Die Geheimnisse hinter einem SAS-EG-Projekt entschlüsseln








Donnerstag, 30.03.2023, 16 h, Hörsaal E-08



Thomas Rüdiger, thomas.ruediger@axa.de

Wie lassen sich Selektionen wiederfinden, die einen oder mehrere Begriffe enthalten?

Im Laufe der Jahre kann sich in einem Unternehmen eine Vielzahl von SAS EG-Projekten und SAS-Programmen ansammeln.

Name	Änderungsdatum	Typ
 dictionary.egp	27.02.2023 19:32	SAS Enterprise Guide Project
 readme.txt	28.02.2023 08:39	Textdokument
 sashelp_class.egp	06.02.2023 19:09	SAS Enterprise Guide Project
 sashelp_class.sas	27.02.2023 19:34	SAS-Datei
 was_anders.sas	27.02.2023 19:36	SAS-Datei



Nicht selten fragt man sich, welche der vorhandenen EG Projekte/SAS Programme gewisse vorgegebene Suchmuster enthalten.

Mögliche Fragestellungen:

- Welche SAS-Programme enthalten eine bestimmte Quelltable?
- Hat sich schon mal jemand mit dem Thema XYZ beschäftigt?

Seit den 1970er Jahren ist der „dir“-Befehl MS-DOS zuständig für das Auflisten von Dateien!

Mit dem nachfolgenden Windows-Programm (MS-DOS) lassen sich EG-Projekte (*.egp) und SAS-Programme (*.sas) in einer Textdatei listen.

```
@echo off
set Suchverzeichnis="C:\ksfe2023\eg_sas_sas"
set Suchtext="Verzeichnis .egp .sas .log"
set NichtText=".sas7bdat .lnk Datentr Volumeseriennummer Datei(en) Verzeichnis(se)"
set Ausgabedatei=C:\ksfe2023\filelisting\filelisting.txt
dir /s /b %Suchverzeichnis%|findstr %Suchtext%|findstr /v
%NichtText%>%Ausgabedatei%
```

Die dir-Option **/s** sorgt dafür, dass Unterverzeichnisse mitbetrachtet werden. Will man nur Dateinamen mit Verzeichnis in der Ausgabe, lässt sich das durch die Option **/b** im dir-Befehl erreichen.

findstr filtert nach den gesuchten Dateiendungen (.egp, .sas, .log) und schließt bspw. SAS-Tabellen (Endung sas7bdat) aus.

Lassen sich SAS-EG-Projekte nur mit dem Enterprise Guide öffnen?

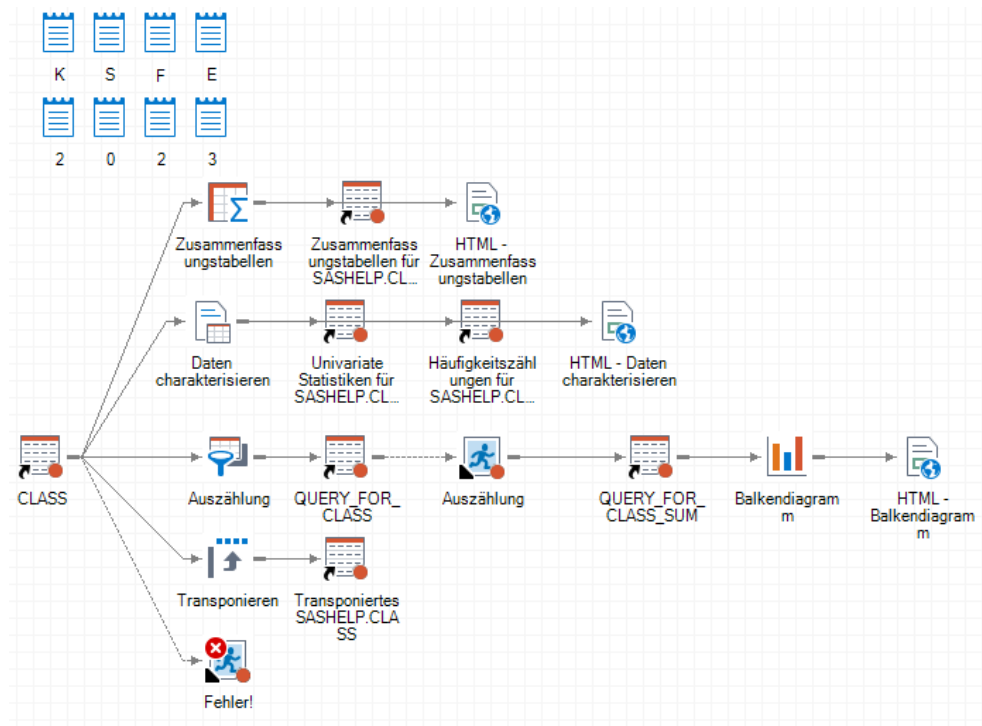
Antwort: Nein!!!!

SAS EG Projekte lassen sich ähnlich öffnen wie SAS-Programme.

Nur: wie????

Wie sieht denn typischerweise ein SAS-EG-Projekt aus?

Ein SAS-EG-Projekt besteht aus Objekten (Programme, Abfragen, Anwendungsroutinen, Ergebnisse, Notizen, ...) in Form eines Projektflusses. Fehler sind **rötlich** gekennzeichnet.



Überraschung: SAS-EG-Projekte sind zip-Archive!

Ein SAS EG-Projekt ist technisch ein zip-Archiv, deren Komponenten sich per unzip-Befehl ausgeben lassen.



In dem dahinterliegenden zip-Archiv befindet sich eine Datei project.xml, über die die gesamte Struktur und Inhalte wie SAS-Logs und SAS-Programme geregelt werden.



SAS-Programme und SAS-Logs können also als Textdateien 1:1 mit den Mitteln der Textanalyse untersucht werden: Text Mining, reguläre Ausdrücke, etc.

Innerhalb des EG-zip-Archivs lassen sich u.a. SAS-Logs und SAS-Programme auslesen!

Name	Größe	Gepackt	Typ	Geändert	CRC32
..			Dateiordner		
CodeTask-Bhh4rb2uhVmT212l	2.825	1.063	Dateiordner		
CodeTask-y4yGU6ZJUBUJWNS2	2.265	912	Dateiordner		
EGTask-3kcSMNMFebM5cAX8	8.606	2.327	Dateiordner		
EGTask-iWxHbU3pQKaTeTCy	8.607	2.441	Dateiordner		
EGTask-LwL5H87OXY3x212D	51.391	9.398	Dateiordner		
EGTask-XpRcgDXsc59JEXbO	10.750	2.623	Dateiordner		
ODSResults	282.616	39.118	Dateiordner		
ProjectLog-vu5xX5qZN95xe5GE	30	32	Dateiordner		
Query-J5PRbbh9PTDE3Y46	2.505	960	Dateiordner		
project.xml	410.720	30.289	XML-Dokument	06.02.2023 19:09	3E2D7F19

CRC=Cyclic Redundant Check (Komprimierungsmethode)

project.xml bildet den roten Faden innerhalb eines SAS-EG-Projekts.

In den Unterverzeichnissen heißen SAS-Programme code.sas, SAS-Logs heißen result.log und HTML-Berichte result.html.

Jedes Element eines SAS-EG-Projekts ist mit Username und Änderungszeitpunkten in project.xml hinterlegt.

<Element>

<Label>Zuletzt übertragener Code</Label>

<Type>CODE</Type>

<Container>ProcessFlowContainer-wAJV7n1cSLBHEcwb</Container>

<ID>Code-3pEPbK9jW4afyfds</ID>

<CreatedOn>638113067123581485</CreatedOn>

<ModifiedOn>638113067590209046</ModifiedOn>

<ModifiedBy>Ruediger, Thomas</ModifiedBy>

<ModifiedByEGID>H78168</ModifiedByEGID>

<ModifiedByEGVer>8.1.2.4594</ModifiedByEGVer>

<HasSerializationError>False</HasSerializationError>

<InputIDs>CodeTask-Bhh4rb2uhVmT212l</InputIDs>

</Element>

Name	Größe	Gepackt	Typ
CodeTask-Bhh4rb2uhVmT212l	2.825	1.063	Dateiordner

Name	Größe	Gepackt	Typ
Log-8xmWBRGAvMyjEIP	2.726	984	Dateiordner
code.sas	99	79	SAS-Datei

code.sas - Editor

```
data Class;
  set SASHELP.Classfix;
  output;
  stop;

run;
proc delete data=Class;
run;
```

CreatedOn und **ModifiedOn** sind in der xml-Datei die Sekunden seit dem Jahr 0 und lassen sich per Formel in ein SAS-datetime-Feld umwandeln:

```
data Zeitpunkt;
  CreatedOn=638113067123581485;
  CreatedOn_SAS=
  intnx('second', '01jan1960:00:00:00'dt
  ,round((CreatedOn-618199776000000000)/1000000));
  format CreatedOn best32. CreatedOn_SAS deufdfd19.;

run;
```

CreatedOn	CreatedOn_SAS
638113067123581440	06Feb2023:18:58:32

Entpackte EG-Projekte und SAS-Programme lassen sich mit regulären Ausdrücken durchsuchen.

```
@echo off
setlocal enableextensions enabledelayedexpansion
set Dateilisting=C:\ksfe2023\filelisting\filelisting.txt
set Ziel_Datei=C:\ksfe2023\textfound\textfound.txt
set Temp_Verzeichnis=C:\temp1
set Zeile=0
set Suche=
set Suche=%Suche% /c:CLASSFIT
set Suche=%Suche% /c:CLASSFIX
if not exist "%Temp_Verzeichnis%" mkdir "%Temp_Verzeichnis%"
set START=%date% %time%

for /f "tokens=*" %%a in (%Dateilisting%) do (
    set /a Zeile += 1
    set Filetype=%%a
    set Filetype=!Filetype:~-3!
    echo #### !Zeile! #### %%a ####>> %Ziel_Datei%
    if "!Filetype!"=="egp" unzip -o "%%a" -d "%Temp_Verzeichnis%"
    if "!Filetype!"=="sas" cp "%%a" "%Temp_Verzeichnis%\"
    findstr /i /r /s %Suche% %Temp_Verzeichnis%\* >> %Ziel_Datei%
    rm -rf %Temp_Verzeichnis%
    mkdir %Temp_Verzeichnis%
    cls
)
)
echo ~~~ Anzahl Dateien: %Zeile% ~~~>>%Ziel_Datei%
echo ~~~ LAUFZEIT: %START% ~~~>>%Ziel_Datei%
echo ~~~ ----- %date% %time% ~~~>>%Ziel_Datei%
notepad %Ziel_Datei%
```



SAS-Programme werden im Codebeispiel in ein temporäre Verzeichnis **kopiert**, SAS-EG-Projekte zusätzlich **entpackt** und mit **findstr** wird in allen Textdateien nach allen gewünschten **Textmustern** analysiert.



```
#### 2 #### C:\ksfe2023\eg_sas_sas\sashelp_class.egp ####
C:\temp1\CodeTask-7us8dq4uAFdktXdO\code.sas:data classfix;
```

Die Suchbegriffe des DOS-Batchprogramms lassen sich als SAS-Tabelle einlesen und für .

Batch

```
set Suche=  
set Suche=%Suche% /c:CLASSFIT  
set Suche=%Suche% /c:CLASSFIX
```



SAS-Tabelle Suchtexte

	Suche	Regex_Suche	Name
1	CLASSFIT	CLASSFIT	CLASSFIT
2	CLASSFIX	CLASSFIX	CLASSFIX

Bei der Zusammenfassung der Ergebnisse lassen sich die Suchbegriffe aus dem Batch recyclen und für eine Macro-Schleife in einer SAS-Tabelle mit SAS-Suchbegriffen und Original-DOS-Suchbegriffen (inkl. regulärer Ausdrücke wie $^[\backslash.]$, $^[0-9]$) hinterlegen.

Für den Fall, dass pro gefundener Textzeile mehrere Suchbegriffe auftreten können, bietet es sich an, pro Suchbegriff zusätzlich eine Boolesche Spalte (1=gefunden 0=nein) mit passendem SAS-Namen zu ergänzen.

Das SAS-Programm liefert die gefundenen Suchergebnisse zu jedem der Dateien aus der DOS-Ausgabe als SAS-Tabelle.

```

%let Textfound_Path=. ;
data Suchtexte(keep=Datei--Text);
  length Datei $40 Type $4 Spalte $8 Text $40;
  infile "&Textfound_Path.\textfound.txt";
  input;
  if _infile_ ^= '' and index(_infile_, '~')=0;
retain Datei;
  if index(_infile_, '#') then Datei=scan(_infile_, 4, byte(32));
  else do;
    Text=left(scan(_infile_, 3, ':'));
    if substr(Text, 1, 1) in ('s' 'e' 'n') and substr(Text, 2, 1) = '
      then Text=left(substr(Text, 3));
      Gross=upcase(Text);
      /*Hier Macro-Schleife der Suchbegriffe aus dem Batch*/
      if index(Gross, "CLASSFIX")^=0 then Spalte="CLASSFIX";
      else if index(Gross, "CLASSFIT")^=0 then Spalte="CLASSFIT";
      else Spalte='';
      if Spalte ^= '' then do;
        if index(upcase(_infile_), '.LOG') then Type='log';
      else if index(upcase(_infile_), '.SAS') then Type='sas';
      else if index(upcase(_infile_), '.SRX') then Type='srx';
      else if index(upcase(_infile_), '.HTML') then Type='html';
      else do;
        put "ERROR: (Type)" +1 _infile_;
      end;
      output;
    end;
  end;
run;

```

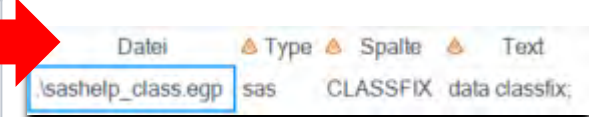
▲	Datei	▲ Type	▲ Spalte	▲ Text
	C:\ksfe2023\leg_sas_sas\sashelp_class.egg	sas	CLASSFIX	set SASHELP.Classfix;
	C:\ksfe2023\leg_sas_sas\sashelp_class.egg	log	CLASSFIX	30 set SASHELP.Classfix;
	C:\ksfe2023\leg_sas_sas\sashelp_class.egg	sas	CLASSFIX	set SASHELP.Classfix;
	C:\ksfe2023\leg_sas_sas\sashelp_class.egg	log	CLASSFIX	30 set SASHELP.Classfix;
	C:\ksfe2023\leg_sas_sas\sashelp_class.egg	sas	CLASSFIX	set SASHELP.Classfix;
	C:\ksfe2023\leg_sas_sas\sashelp_class.egg	log	CLASSFIX	30 set SASHELP.Classfix;

Über die Trefferliste erhält man alle Dateien, die die ausgewählten Suchbegriffe enthalten.

Gefundener Text

```
#### 2 #### C:\ksfe2023\eg_sas_sas ####  
C:\temp1\CodeTask-7us8dq4uAFdktXdO\code.sas:data classfix;
```

SAS-Tabelle Suchtreffer



Datei	Type	Spalte	Text
.\sashelp_class.egg	sas	CLASSFIX	data classfix;

Aus dem dir-Befehl (ohne die Option /b“) ergeben sich weitere Informationen:



- Erstelzeitpunkt (Datum/Uhrzeit)
- Dateigröße (Bytes)
- Datei-Owner (Username)

Spalte	Wert
Datei	.\sashelp_class.egg
Username	RUEDIGER Thomas
Jahr	2018
Datum_Zeit	12Sep2018:11:55:34
Bytes	19.118
Type	sas
Spalte	CLASSFIX
Suche	CLASSFIX
Anzahl	3
Text	data classfix;

Über eine Auszählung jeder der Suchbegriffe erhält man eine Übersicht auch nicht gefundener Suchbegriffe.

```
data Auszaehlung;
  length Spalte $8;
  do Spalte="CLASSFIT" , "CLASSFIX";
    output;
  end;
run;

proc sql;
  create table Auszaehlung as
  select distinct a.Spalte,
  case
  when b.Anzahl is not null then b.Anzahl
  else 0
  end as Anzahl format=commax32.
  from Auszaehlung a
  left join
  (select distinct Spalte, count(*) as
Anzahl from Suchtexte group by Spalte) b
  on a.Spalte=b.Spalte
  order by Anzahl desc, Spalte;
quit;
```

 Spalte	 Anzahl
CLASSFIX	6
CLASSFIT	0



Danke fürs Zuhören!



Thomas Rüdiger
AXA Konzern AG
thomas.ruediger@axa.de