

# Gestalterische Elemente für das Erstellen von SAS Portalen auf Linuxservern

Thomas Rüdiger  
 AXA Konzern  
 Colonia-Allee 19-21  
 51067 Köln  
 thomas.ruediger@axa.de

## Zusammenfassung

SAS Portale auf Linuxservern bieten eine zentrale Anlaufstelle für unternehmensrelevante Kennzahlen und Zeitreihen. Die zugehörigen Portal-Daten lassen sich automatisiert per Linuxbatch (at, cron) und über PROC TEMPLATE im Corporate Design bereitstellen. Über HTML-Formularfelder (Auswahlboxen, Textfelder) lassen sich Berichte filtern. Die Anwender können durch den SAS-Entwickler für einzelne Portalseiten berechtigt bzw. nicht berechtigt werden. Die hier dargestellten Daten stammen aus einer aus SASHELP.HEART abgeleiteten SAS Tabelle mit Patientenummer, Behandlungszeitraum und ins Deutsche übersetzte Texte in Textfeldern.

**Schlüsselwörter:** Portal, Linux, PROC FREQ, PROC MEANS, PROC TEMPLATE, HTML, Java

## 1 Allgemeine Daten

### 1.1 SAS Makrovariablen

Häufiger verwendete und/oder veränderbare Werte (Abteilungskürzel, Links, JavaScripte) lassen sich in einer zentralen SAS Tabelle hinterlegen und als globale SAS Makrovariablen abrufen.

```
%macro MacVar (Name, Wert, Desc);
  %if %length(&Name.)^=0 and %length(&Wert.)^=0 %then %do;
    Name="_%upcase(&Name.)";
    Value="&Wert.";
    Gross=upcase(Value);
    if      index(Gross, '.JPG')^=0 then Type='jpg';
    else if index(Gross, '.PNG')^=0 then Type='png';
    else if index(Gross, '.GIF')^=0 then Type='gif';
    else if index(Gross, '.JS')^=0  then Type='js';
    else if index(Gross, '.HTML')^=0 then Type='html';
    else if index(Gross, '.HTM')^=0 then Type='htm';
    else Type='';
    Desc="&Desc.";
  output;
%end;
```

```
%mend MacVar;
data &Tab_Globale_Macrovariablen.(compress=no label="Allgemeine Re-
reporting-Parameter" keep=Name--Type);
  attrib
    Name length=$32 label="Name"
      Desc length=$80 label="Beschreibung"
    Value length=$200 label="Wert"
    Type length=$4 label="Typ"
  ;
%MacVar(Session_Timeout_Sec,1800,Session Timeout);
run;

proc sort noequals nodupkey data=&Tab_Globale_Macrovariablen.;
  by Name;
run;
```

## 1.2 Ausgabeformate

Ausgabeformate wie Bildschirm, Excel oder pdf (SAS ODS Destinations) lassen sich über eine Referenztabelle später als Auswahlbox anzeigen.

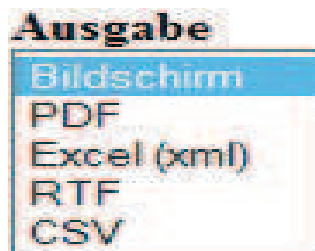


Abbildung 1: Ausgabeformate

```
%macro Destina-
tion(VarNum,ODS,Select_Option_Value,Select_Option_Anzeige
,File_Extension,Content_Type,Graph_Output);
  VarNum=&VarNum.;
  ODS=lowercase("&ODS.");
  Select_Option_Value="&Select_Option_Value.";
  Select_Option_Anzeige="&Select_Option_Anzeige.";
  File_Extension=lowercase("&File_Extension.");
  Content_Type="&Content_Type.";
  %if %length(&Graph_Output.)^=0 and %upcase(&Graph_Output.)^=N %then
%do;
  Graph_Output='1';
%end;
%else %do;
  Graph_Output='';
%end;
  output;
%mend Destination;

data &Tab_Destinations.(compress=no label="Destinations");
  attrib
    VarNum length=3 label="Destination Nr."
    ODS length=$10 label="SAS ODS Output"
    Select_Name length=$8 label="Bezeichnername"
```

```

Select_Option_Value length=$6 label="Absendewert"
Select_Option_Anzeige length=$20 label="Eintrag der Auswahl-
liste"
    File_Extension length=$4 label="Datei-Endung"
    Content_Type length=$80 label="Content-Type"
    Graph_Output length=$1 label="1=Graphik-Output"
    ;

    retain VarNum 0 Select_Name "_Ausgabe";
    %Destination(1,html,screen,Bildschirm,,j);
    %Destination(2,pdf,pdf,PDF,pdf,%str(application/pdf),j);
    *Destination(3,excel,excel,Excel,xls,%str(application/vnd.ms-
excel),n);
        %Destination(4,html,excel,%str(Excel
(xml)),xls,%str(application/vnd.ms-excel),n);
        %Destination(5,rtf,rtf,RTF,rtf,%str(application/rtf),j);
        %Destination(6,csv,csv,CSV,csv,%str(application/vnd.ms-
excel),n);

*Destination(7,powerpoint,ppt,Powerpoint,pptx,%str(application/vnd.o
penxmlformats-officedocument.presentationml.presentation),j);
run;

proc sort noequals nodupkey data=&Tab_Destinations.;
    by VarNum;
run;

```

### 1.3 Benutzerinfo über Linuxbefehle

SAS kann mithilfe von Linux-Systembefehlen Usereigenschaften zur Userid ermitteln. Dazu gehören:

- Liste mit Userids einer Linux-Gruppe: `getent group | grep -i {Linux-Gruppe}`
- Abteilungs-ID (daraus ableitbar Abteilungsname): `ldapsearch -x uid={User-Id} | grep -i gidNumber | cut -d ':' -f2`
- Username, daraus i.d.R. ableitbar die Emailadresse: `getent passwd {User-Id} | cut -f5 -d ':'`

### 1.4 Berichtsverwaltung

Sämtliche Portalseiten lassen sich mit Beschreibungen, Kürzel (z.B. 3-Letter-Code) und Links über eine Referenzdatei verwalten. Das einfach gehaltene Kürzel wird für die Benutzerberechtigung pro Portalseite verwendet.

```

%macro TaskData(Task_Visible=,Task_Path=,Stored_Process_HTML=
,Stored_Process_Report=,Task_ShortCut=,Task_Label=,Task_Link=);
    %if %length(Task_Link.)>0 %then %do;
        Task_No=Task_No+1;
        %if &Task_Visible.=1 or %upcase(&Task_Visible.)=J %then %do;
            Task_Visible="1";

```

```
        %end;
        %else %do;
            Task_Visible="";
        %end;
    Task_Path="&Task_Path.";
    Stored_Process_HTML="&Stored_Process_HTML.";
    Stored_Process_Report="&Stored_Process_Report.";
    Task_ShortCut=compress("%upcase(&Task_ShortCut.)");
    Task_Label="&Task_Label.";
    Task_Link="&Task_Link.";
    output;
%end;
%mend TaskData;

data &Tasktabelle.(compress=no label="Portal-Tasks");
    attrib
        Task_No length=3 label="Portalanwendung Nr."
        Task_Visible length=$1
label="Link sichtbar 1=ja sonst=versteckt"
        Task_Path length=$80
label="Physisches Verzeichnis der Portalanwendung"
        Stored_Process_HTML length=$20
label="Stored Process (Auswahl)"
        Stored_Process_Report length=$10
label="Stored Process (Bericht)"
        Task_ShortCut length=$4
label="Kürzel Portalanwendung"
        Task_Label length=$80
label="Beschreibung der Portalanwendung"
        Task_Link length=$400
label="Link zur Portalanwendung"
    ;
    retain Task_No 0;

    %TaskData(Task_Visible=j
        ,Task_Path=Test_Heart_&sysuserid.
        ,Stored_Process_HTML=Auswahl
        ,Stored_Process_Report=Bericht
        ,Task_ShortCut=HRT
        ,Task_Label=%str(Dashboard Heart-Daten)

,Task_Link=%bquote(&ABNA_Link.?&Param.Test_Heart_&sysuserid.%2FSchul
ung%28StoredProcess%29));

run;

proc sort noequals nodupkey data=&Tasktabelle.;
    by Task_No;
run;
```

## 1.5 Benutzerberechtigung pro Portalseite

Für jede Portalseite (Task\_ShortCut) lässt sich individuell pro User (User\_Id) eine Berechtigungsstufe (Task\_Auth) über eine SAS-Tabelle hinterlegen. Task\_Auth=0 kann z.B. alle Reportingdetails bedeuten.

Praktischerweise lässt sich für die Datenpflege aufgrund der wenigen benötigten Spalten auf Linuxservern eine csv-Datei zum Einlesen nach SAS hinterlegen.

## 2 Portal-Daten

Als Beispieltabelle wird eine aus **SASHELP.HEART** abgeleitete SAS mit Übersetzung der Textfelder ins Deutsche genommen.

PATNO	BDAT	Status	DeathCause	Sex	Chol_Status	BP_Status	Weight_Status	Smoking_Status	AgeCHDdiag	Age4Start	Height
1	08.01.2018	lebendig		männlich	Hoch	Hoch	Übergewicht	Nichtraucher		33	14
2	08.07.2003	lebendig		männlich	Hoch	Normal	Normal	Schwer (16-25)		48	18

**Abbildung 2:** Beispieldaten

Sämtliche verwendete Daten müssen beim Berichtsaufwurf für den SAS Stored Process User lesbar sein. Auf Linuxservern hat der Stored Process User die Funktion von „Other User“ mit einer eigenen Userid. Zu den Datentabellen gehören typischerweise:

- Detaildaten oder Kennzahlen-Tabellen für die jeweils benötigte Zeitperiode (Monat, Woche, Tag)
- Meta-Tabelle mit Infos zu Datenständen
- Meta-Tabelle mit den Ausprägungen zu kategoriellen Variablen (den späteren Auswahlboxen)
- Ggf. Meta-Tabelle zur Kapitelauswahl oder Kennzahldefinition

### 2.1 Datentabelle

Der Aufbau eines SAS Portal richtet sich nach den nachfolgenden Feldtypen einer SAS Tabelle

- Keyvariablen (Primär-/Fremdkey, numerisch/Text): Werden i.d.R. nur bei Sicht auf Detaildaten benötigt
- Kategorielle Felder (Class-Felder, numerisch/Text): Felder mit wenigen disjunkten Ausprägungen
- Statistische Felder (Summe/Anzahl/Beträge, nur numerisch)
- Datums-/Uhrzeitfelder (in SAS numerisch)
- Sonstige Textfelder (Hinweistexte, etc.)

### 2.2 Datenstände

Datenstände sind über Metadaten von SAS/Oracle/DB2/.. oder über Datumfelder (Datenstatus) ermittelbar.

Datenstände lassen sich auf der Portalseite als Auswahlbox anzeigen und Zeiträume per Javascript auswählen.



Abbildung 3: Datumsauswahl

### 2.3 Kategorielle Felder – PROC FREQ

Die Ausprägungen kategorieller Felder lassen sich über die SAS Prozedur FREQ in eine SAS Tabelle schreiben und auf der Portalseite als HTML-Auswahlfelder (select-Felder) mit Einzelausprägungen darstellen.

```
ods _all_ close;
ods output OneWayFreqs = &Tab_Meta_Class.;
proc freq data=&Datenbasis.(keep=&Var_Meta_Class.);
    tables &Var_Meta_Class./missing nocum nopercnt;
    format _all_;
run;ods output close;ods listing;
```

Im Vorfeld der Portalanwendung lässt sich über die entsprechende Datentabelle festlegen, welches dieser Felder sich über Multiple Choice oder Single Choice auswählen lassen soll und ob es abhängige Beziehungen gibt (Beispiel: Todesursache nur anzeigen, wenn Status=,verstorben‘.

VarNum	Select_Name	Display	Select_Label	Multiple	Select_Op- oup_Value	Select_Op- n_Value	Select_Op- n_Anzeige	Show_Value	Select_Name _Child
1	3 _STATUS	inline	Status			01-lebendig	lebendig	1	
2	3 _STATUS	inline	Status			02-verstorben	verstorben	1	_DEATHCAUSE
3	4 _DEATHCAUSE	none	Todesursache	1		01-Krebs	Krebs	1	
4	4 _DEATHCAUSE	none	Todesursache	1		02-Hirngefäßl...	Hirngefäßlakra...	1	
5	4 _DEATHCAUSE	none	Todesursache	1		03-Koronare H...	Koronare Herz...	1	
6	4 _DEATHCAUSE	none	Todesursache	1		04-Andere Erkr...	Andere Erkrant...	1	
7	4 _DEATHCAUSE	none	Todesursache	1		05-Unbekannt	Unbekannt	1	
8	4 _DEATHCAUSE	none	Todesursache	1		99-	K.A.	1	
9	5 _SEX	inline	Geschlecht	1		01-männlich	männlich	1	
10	5 _SEX	inline	Geschlecht	1		02-weiblich	weiblich	1	

Abbildung 4: Kategorielle Felder

### 2.4 Kennzahlentabelle – PROC MEANS

Für schnell sichtbare Ergebnisse lassen sich mit PROC MEANS zu allen kategoriellen Feldern (class) auf verschiedenen Aggregationsebenen (Output: \_TYPE\_) zu ausgewählten numerischen Feldern (var).

## 2.4 Berichts-Kapitel

Für schnell sichtbare Ergebnisse lassen sich mit PROC MEANS zu allen kategoriellen Feldern (class) auf verschiedenen Aggregationsebenen (Output: `_TYPE_`) zu ausgewählten numerischen Feldern (var).



Abbildung 5: Berichts-Kapitel

## 3 Portal-Seite

Jede Portalseite benötigt i.d.R. 3 SAS Stored Processes

- Frameset (Startseite/Gesamtsicht)
- HTML-Formular (SAS-put-Anweisungen aus Metadaten zur Portalseite)
- SAS Report (proc report, proc sgplot, etc.)

Jede Portalseite ist auf einwandfreie Interaktivität zwischen HTML-Formular und SAS Bericht zu prüfen.

### 3.1 Frameset

Der aus dem SAS EG erstellte Stored Process zum Frameset verweist auf die Stored Processes zum HTML-Formular und SAS Report und legt die Größenordnung der zugehörigen HTML-Frames fest. Die von SAS vorgegebene Filereferenz `_webout` wird in diesem und allen nachfolgenden Stored Processes verwendet und kann testweise im SAS EG auf eine physische Datei verweisen.

### 3.2 HTML-Formular

Das HTML-basierte Auswahlformular besteht aus

- Frameset (Startseite/Gesamtsicht)
- HTML-Text
 

```
<html><head>...</head><body>...<table>...</table>...</body></html>
```
- Darin ein HTML-Formular für die an SAS übergebenen Parameter
 

```
<form action="do" target="main" name="fname">...</form>
```
- Ggf. nützliche JavaScript-Funktionen:
  - Ein- und Ausblenden von kaskadierenden Elementen (Sub-Auswahlboxen)
  - Datumsauswahl per Kalender

Der Stored Process zum HTML-Formular besteht aus put-Anweisungen in HTML-Sprache. Datenstände, Kapitel, Auswahlboxen, Ausgabeformate in HTML lassen sich über die vorgestellten SAS-Metatabellen erstellen.

Zusätzlich lassen sich in der put-Anweisung verwendete JavaScripte (Objekte sichtbar machen, Kalenderauswahl) sowie die „Sende“-/„Rückgängig“-Button ergänzen.

### **3.3 SAS Report**

Die Werte aus dem HTML-Formular (Parameter/Datenfilter) lassen sich für die Berichterstattung aus den globalen Macrovariablen (Dictionary.Macros=SASHELP.VMacros) auslesen und als Parameter für AusgabeprozEDUREN wie PROC REPORT oder PROC PRINT oder "data \_null\_;file \_webout;put ...;run;"-Ausgaben verwenden.

Style-Elemente der Reports sind zentral über PROC TEMPLATE an das standardmäßig über css-Dateien festgelegte Corporate Design anpassbar.