

# SAS/AF® on a shoestring: Rapid prototyping for bureaucratic persuasion

19 November 2008

Daniel Stripinis  
Peter Tarassoff

# Bureaucratic persuasion

**Need:** Labour-saving application for internal staff and external respondents to use in capturing data.

**Example:** Traditional development of another application with similar functionality had been estimated at several \$100K over two years.

**“Logic”:** “No.” to perceived “sunk cost” of \$2,400 for a 1-year SAS/AF licence, assuming some few \$100K more needed for completion here, too.

**Solution:** Prototype with a SAS/AF evaluation licence!

# Rapid prototyping

SAS/AF evaluation licences have built-in durations of 15 days, hence the need for rapid prototyping.

Indeed, if one were to want to do a thorough evaluation of SAS/AF, one would probably need to devise an extremely detailed evaluation plan in advance, perhaps to the extent of developing SCL code without SCL ...

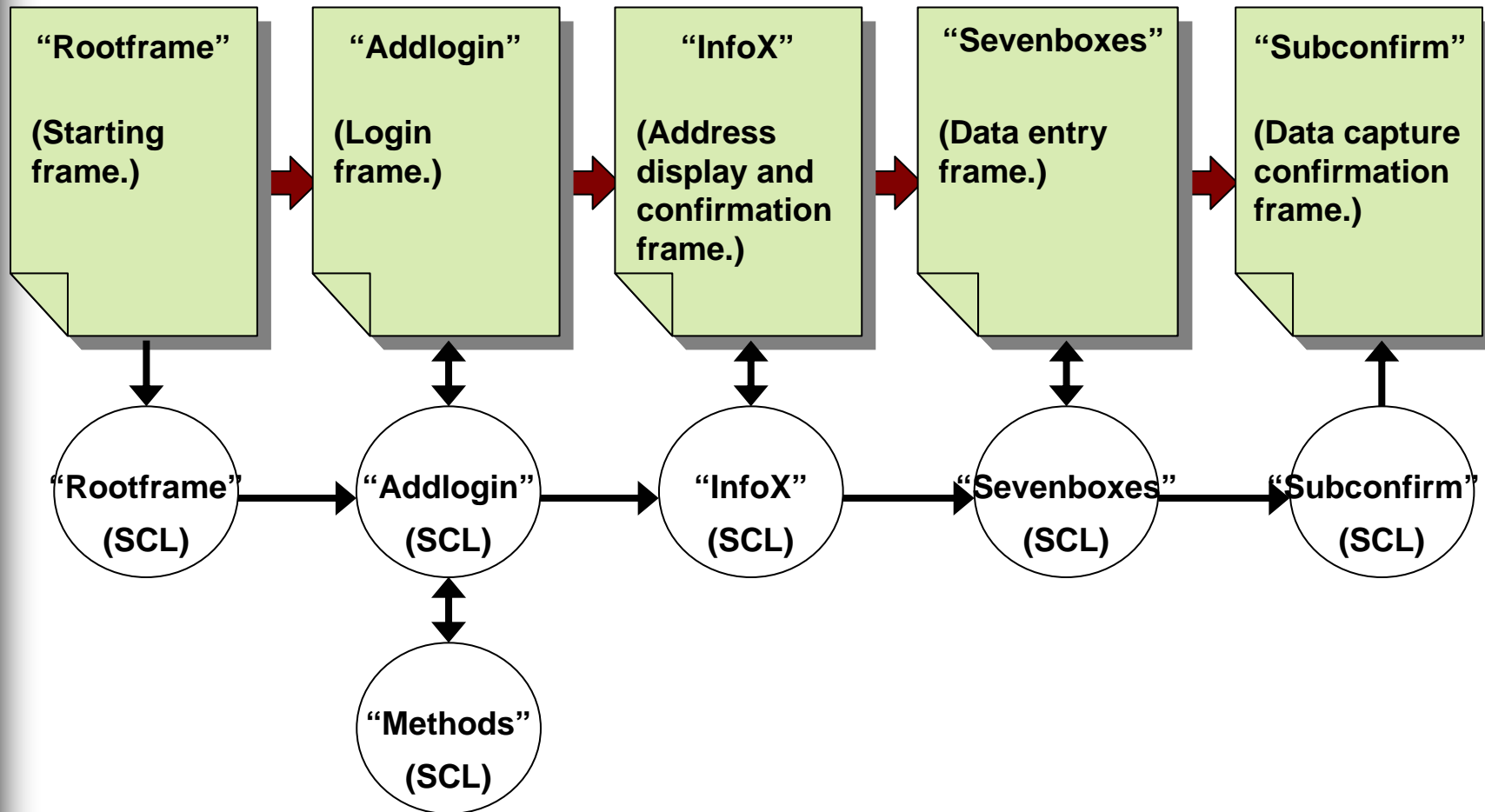
# The EAHOR application

## High-level requirements (as prototyped):

1. Client logs in using id number and password;
2. We display client id number, name, and address, which the client can confirm;
3. We display a data entry form where the client can enter data on hazardous occurrences, employee numbers, and hours worked. As data are entered, they are validated;
4. Client can submit validated data, which are written to a SAS file; and
5. We confirm that data have been submitted.

*“Come out of the desert of ignorance to the OASUS of knowledge”*

# EAHOR: 5 AF frames run with 6 SCL programs



# EAHOR: 5 AF frames and 6 SCL programs

Name	Size	Type
Addlogin	6.6KB	Frame
Infox	24.1KB	Frame
Rootframe	2.2KB	Frame
Sevenboxes	26.8KB	Frame
Subconfirm	3.1KB	Frame
Addlogin	9.0KB	Scl
Infox	5.8KB	Scl
Methods	1.1KB	Scl
Rootframe	1.3KB	Scl
Sevenboxes	14.1KB	Scl
Subconfirm	1.0KB	Scl

# Rootframe (going to log in)

```
GoLogIn:  
  call DISPLAY('Fjidguiu.Webeahor.Addlogin.Frame',",");  
RETURN;
```

Web EAHOR

Go Log In



# Addlogin (initialising data)

## Web EAHOR

Employer Identification Number

Employer Password

Log In

```
INIT:
```

```
/* Specify the SAS library. */
```

```
RC=LIBNAME("FJ_AD",  
"U:\LAB-TRA\OPS\OHSFP-  
SSTPI\GLOBAL\RA_SAS_Files\SAS_AF_Project\SA  
S_Data");
```

```
/* Load the data into memory. */
```

```
SUBMIT CONTINUE;  
SASFILE FJ_AD.FJ_WEB_EMPLOYERS.DATA  
LOAD;  
ENDSUBMIT;
```

```
...
```



# Addlogin (overriding methods)

## Web EAHOR

Employer Identification Number

Employer Password

Log In

```
...  
  
/* Override the "_onValueChanged" method. */  
  
ClientNumber._setinstancemethod('_onValueChanged',  
    'Fjidguiu.Webeahor.Methods.SCL','onKey');  
ClientPassword._setinstancemethod('_onValueChanged',  
    'Fjidguiu.Webeahor.Methods.SCL','onKey');  
  
/* Override the "_onreturnkey" method. */  
  
ClientNumber._setinstancemethod('_onreturnkey',  
    'Fjidguiu.Webeahor.Methods.SCL','onreturnkey');  
ClientPassword._setinstancemethod('_onreturnkey',  
    'Fjidguiu.Webeahor.Methods.SCL','onreturnkey');  
  
...
```

# Methods (used by Addlogin)

```
/* Assorted SCL Methods (thanks to Greg McLean) */  
  
_SELF_=_SELF_;  
onkey:  
  METHOD;  
  _SELF_.objectLabel();  
  ENDMETHOD;  
  
onreturnkey:  
  METHOD;  
  call SYMPUT("ENTER_HIT","YES");  
  ENDMETHOD;  
  
onvaluechanged:  
  METHOD;  
  _SELF_.objectLabel();  
  ENDMETHOD;
```

# Addlogin (checking id number)

## Web EAHOR

Employer Identification Number

Employer Password

Log In

```
ClientNumber:  
ClientPassword:
```

```
/* Ensure that ClientNumber is 6-digit number. */
```

```
if ClientNumber.text=
```

```
    PUTN(ClientNumber.text,'Z6.')
```

```
    AND 6=LENGTH(ClientNumber.text) then do;
```

```
    ClientPassword._UNGRAY();
```

```
    if ClientPassword.text ne "" then
```

```
        LoginSubmit._UNGRAY();
```

```
    else
```

```
        LoginSubmit._GRAY();
```

```
end;
```

```
else do;
```

```
    ClientPassword.enabled='no';
```

```
end;
```

```
if (SYMGET("ENTER_HIT")="YES") then do;
```

```
    if (ClientNumber.text=PUTN(ClientNumber.text,'Z6.')
```

```
        AND 6=LENGTH(ClientNumber.text)
```

```
        AND ClientPassword.text ne "") then
```

```
    LINK LoginSubmit;
```

```
    call SYMPUT("ENTER_HIT","NO");
```

```
end;
```

```
call SYMPUT("ENTER_HIT","NO");
```

```
RETURN;
```

# Addlogin (submitting id no.)

## Web EAHOR

Employer Identification Num

Employer Password

Log In

LoginSubmit:

```
PARM_ClientNumber =ClientNumber.text;  
PARM_ClientPassword =ClientPassword.text;
```

```
/* Validate the client number vs. our data. */
```

```
C_Table=OPEN("FJ_AD.FJ_WEB_EMPLOYERS");
```

```
RC=LOCATEC(C_Table,VARNUM(C_Table,"EstNum6"),  
PARM_ClientNumber,"A");
```

```
IF (RC=0) THEN DO;
```

```
/* place-holder */
```

```
END;
```

```
ELSE DO;
```

```
/* If they pass validation, bring up the */
```

```
/* "INFOX" frame. */
```

```
call DISPLAY('Fjidguiu.Webeahor.InfoX.Frame',  
PARM_ClientNumber,  
C_Table);
```

```
call EXECCMD('end;');
```

```
END;
```

```
RETURN;
```

# InfoX (getting SAS data)

## Employer's Annual Hazardous Occurrence

Please review your employer name and Canadian head office address. If either is incorrect, please click on "Make Corrections", correct the information, and click on "Save Corrections". If both are already correct, then please click on "Done".

Employer Identification Number

Employer Name

Street Number

Unit / Suite / Apt.

Street Name





```

...
/* Get the ClientNumber's record from the      */
/* WEB_EMPLOYERS dataset.                      */

CALL SET(C_Table);

GetRec=FETCHOBS(C_Table,LOCATEC(C_Table,
VARNUM(C_Table,"EstNum6"),
PARM_ClientNumber,"A"));

/* Close the dataset. */

CloseIt=CLOSE(C_Table);
...

/* Get and display data from the record. */

EmployerName_O.text=EstName;
StreetNumber_O.text=ST_Num;
...

```

# InfoX (confirming information)

Country

  
  
  
Postal Code  
  
  

ConfirmNameAddress:

```
/* If the user confirms the name and address */  
/* address, bring up the "SEVENBOXES" frame. */  
  
call display('Fjidguiu.Webeahor.Sevenboxes.Frame',  
            PARM_ClientNumber,  
            EstName,  
            ST_Num,  
            Unit_Suite_Apt,  
            ST_Name,  
            ST_Type,  
            ST_Direction,  
            PO_Box,  
            City_Town,  
            Province,  
            Country,  
            P_Code);  
  
call EXECCMD('end;');  
RETURN;
```



# Sevenboxes (initialising)

	[This Year]
Number of Disabling Injuries	<input type="text"/>
Number of Deaths	<input type="text"/>
Number of Minor Injuries	<input type="text"/>
Number of Other Hazardous Occurrences	<input type="text"/>
Total Number of Employees (FTE)	<input type="text"/>
Number of Office Employees (FTE)	<input type="text"/>
Total Number of Hours Worked	<input type="text"/>
	[This Year]

```

INIT:
...

/* Reset missing values, so that decimal text */
/* entry controls are *not* initialized with "." */
MV=OPTSETC("MISSING",");

/* Initialise, disable or grey some objects. */
ClientNumber.text=PARM_ClientNumber;
ClientNumber.enabled="no";
...
DataSubmit._GRAY();
...
call SYMPUT("ENTER_HIT","NO");
...
NoDI_CY.enabled="no";
NoD_CY.enabled="no";
...

if (Year_CY.text="") then Year_CY._CURSOR();
RETURN;

```

*“Come out of the desert of ignorance to the OASUS of knowledge”*



# Sevenboxes (checking year)

	[This Year]
Number of Disabling Injuries	<input type="text"/>
Number of Deaths	<input type="text"/>
Number of Minor Injuries	<input type="text"/>
Number of Other Hazardous Occurrences	<input type="text"/>
Total Number of Employees (FTE)	<input type="text"/>
Number of Office Employees (FTE)	<input type="text"/>
Total Number of Hours Worked	<input type="text"/>
	[This Year]

```

Year_CY:
/* Ensure that Year_CY is a 4-digit number, */
/* that it is prior to the year of data entry, and */
/* that it is no earlier than 2000. */
if Year_CY.text=
    PUTN(Year_CY.text,'Z4.')
    AND 4=LENGTH(Year_CY.text)
    AND (Year_CY.text LT YEAR(TODAY()))
    AND (Year_CY.text GE "2000")
then do;
    NoDI_CY.enabled="yes";
    NoD_CY.enabled="yes";
    ...
end;
else do;
    NoDI_CY.enabled="no";
    NoD_CY.enabled="no";
    ...
end;
RETURN;
    
```

# Sevenboxes (checking data)

Number of Disabling Injuries

Number of Deaths

Number of Minor Injuries

Number of Other Hazardous Occurrences

Total Number of Employees (FTE)

Number of Office Employees (FTE)

Total Number of Hours Worked

```

/* Check the entered data, before enabling the */
/* "Data Submit" button. */
NoDI_CY:
...
TotalHours_CY:
If
  /* All numbers must be GE 0. */
  (NoDI_CY.text GE 0
  AND NoD_CY.text GE 0
  ...
  AND TotalHours_CY.text GE 0)

  /* OfficeFTEs_CY must be LE TotalFTEs_CY. */
  AND (OfficeFTEs_CY.text LE TotalFTEs_CY.text)

  /* TotalHours_CY must be GE 1000*TotalFTEs_CY */
  /* AND LE 3000*TotalFTEs_CY. */
  AND (TotalHours_CY.text GE 1000*TotalFTEs_CY.text
  AND
  TotalHours_CY.text LE 3000*TotalFTEs_CY.text)
then
  DataSubmit._UNGRAY();
else
  DataSubmit._GRAY();
RETURN;

```

# Sevenboxes (converting data)

Submitting Officer's Name and Title

Telephone

Submit Data

DataSubmit:

```
/* Capture the data first as SCL variables, then */  
/* as SAS dataset variables. Also capture a      */  
/* datetime stamp.                               */  
/* Capture the data as SCL variables. */  
PM_Year_CY  =Year_CY.text;  
...  
PM_TotHrs_CY =TotalHours_CY.text;  
  
TY=PUT(YEAR(TODAY()),z4.);  
TM=PUT(MONTH(TODAY()),z2.);  
TD=PUT(DAY(TODAY()),z2.);  
  
PM_SubDate  =TRIM(TY)||TRIM(TM)||TRIM(TD);  
PM_SubTime  =PUT(TIME(),HHMM5.);/*###*/  
/*NB: Format is FUBAR when writing to table. */  
...
```

# Sevenboxes (storing data)

Submitting Officer's Name and Title

Telephone

Submit Data

```
/* If the sevenboxes dataset DOES (now) exist, */  
/* append the current session's record to it. */  
if sevenbox_exist then do;  
  tableid=OPEN('FJ_AD.SevenBox','U');  
  call putvarc(tableid,varnum(tableid,'EstNum6'),PARM_ClientNumber);  
  ...  
  call putvarn(tableid,varnum(tableid,'NoDI_CY'),PM_NoDI_CY);  
  ...  
  call putvarn(tableid,varnum(tableid,'TotHrs_CY'),PM_TotHrs_CY);  
  if (append(tableid,'noinit')) then  
    _msg_=sysmsg();  
  rc=CLOSE(tableid);  
end;  
  
call display('Fjidguiu.Webeahor.SubConfirm.Frame');  
call EXECCMD('end;');  
RETURN;
```

# Subconfirm (confirming submission)

Web EAHOR

Employer's Annual Hazardous Occurrence Report

Thank you for submitting your data!

# Rapid prototyping with SAS/AF: Observations

Using the available documentation, as well as some sample catalogs from Greg McLean, a basic knowledge of SAS/AF was not terribly difficult to acquire and apply within 15 days. However, without those sample catalogs, it would have been another story.



# Rapid prototyping with SAS/AF: Observations

To make rapid prototyping with SAS/AF easier and less risky, evaluation licences for more than 15 days would help. Likewise, if example catalogs were bundled with SAS/AF. Finally, we noticed that SAS/AF was *not* covered in “SAS Help and Documentation” when SAS/AF was not yet installed. If it had been included, we could have prepared sooner. Also, covering more products in “SAS Help and Documentation” may help expose potential clients to those products ...

*“Come out of the desert of ignorance to the OASUS of knowledge”*



# Bureaucrats persuaded?

The prototype may have helped convince people that the intended project could bypass the official three-year development process. People do seem more interested - it has become much more real and feasible and they are willing to at least discuss the possibility that we can get this done in a couple of months.

# Further reading

Barnes Nelson, G.S., “Fundamentals of FRAME: A Tutorial for Building Applications.” Paper 179.

Boling, John C., “Introduction to FRAME Entries in SAS/AF Software.” Paper 145.

Imken, Bernd E., “The Q & A’s of SAS/AF Development.” Paper 027-29, SUGI 29, 2004.

McLean, Greg, “‘That Can’t Be SAS!’: Starting SAS/AF Applications.” Paper 047-30, SUGI 30, 2005.

McLean, Greg, “I Shall RETURN – Key Feedback in SAS/AF.” Paper 111-2008, SAS Global Forum 2008.

McLean, Greg, “Picture Perfect: Centering Frames in SAS/AF.” Paper 009-2008, SAS Global Forum 2008.

# Further reading (continued)

McLean, Greg, "Super Control Language!: The Power of SAS SCL." Statistics Canada SAS Users Group, November 6, 2008.

SAS Institute, *Getting Started with SAS/AF and Frames*, 2006.

SAS Institute, *SAS/AF 9.1 Procedure Guide*, 2004.

SAS Institute, *SAS Component Language 9.1 Reference*, 2004.

SAS Institute, *SAS Guide to Applications Development*, 2d ed., 2004.

Wilson, Steven A., "Getting Started with SAS/AF Software." Paper 193-27, SUGI 27, 2002.

*"Come out of the desert of ignorance to the OASUS of knowledge"*