

Macro to manage the PCSA

Claude Guyot
PhUSE 2009 – Basel
Paper CS02

Agenda

- Standardization of the PCSA definition
 - What are the PCSA?
 - Who defines the PCSA in Sanofi-Aventis?
- Macro to derive the PCSA
 - Objectives of the macro
 - Parameters of the macro
 - Two system datasets
 - PCSA conditions
 - Macro-variables used in the macro
- CALL SET Routine and FETCH Function

Standardization of the PCSA definition

■ What are the PCSA?

■ Potentially Clinically Significant Abnormalities

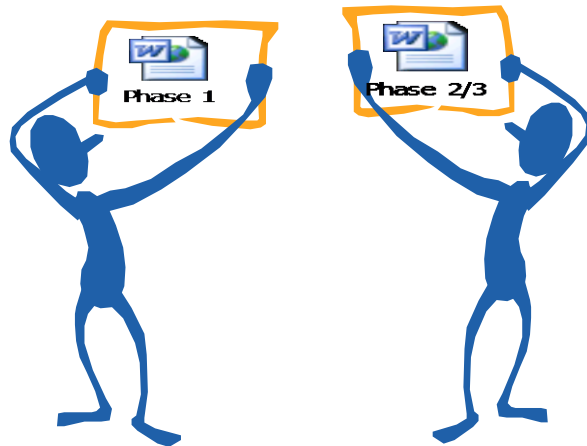
■ Definition

- A Potentially Clinically Significant Abnormality is a major deviation in comparison with the normal limits for:
 - Laboratory data (clinical chemistry and hematology)
 - Electrocardiogram parameters
 - Vital signs
- A PCSA leads to a close surveillance in the clinical studies

Standardization of the PCSA definition

■ Who defines the PCSA in Sanofi-Aventis?

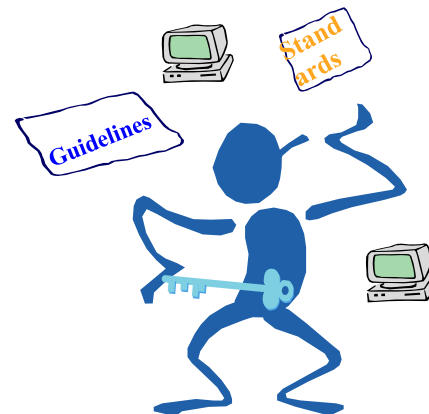
Pharmacovigilance (PV)



Simplification of the criteria:

- Only one unit per parameter
- Homogeneization of the labels

Biostatistics & Programming (B&P)



How to enter PCSA in the database?

How to standardize the PCSA?

How to analyse the PCSA?

From baseline or not?

How to report PCSA in CSR?

Macro to derive the PCSA

■ Objectives of the macro

- Derive the PCSA variables in the Analysis DataSets (ADS) in a standard way for Laboratory Data, ECG and Vital Signs
 - Label of PCSA
 - Flags
 - Status at baseline
- For laboratory data, flag the out of range values
- Allow to re-run and derive the PCSA variables with previous versions of PCSA definition

Macro to derive the PCSA

■ Parameters of the macro

```
%st_saflags(  
    domain      = <domain: LB or EG or VS>  
    ,phase      = <phase: 1 (for phase 1) or 3 (for phase 2/3)>  
    ,dsin       = <input dataset>  
    ,dsout      = <output dataset>  
    ,dsoutpcsa  = <output dataset to store the PCSA definitions of the study parameters>  
    ,dspcsa     = <system dataset for PCSA definitions>  
    ,dsref      = <system dataset to link macro-variables and values>  
    ,cutoff     = <cut-off date to select version of the PCSA>  
    ,lnrefer    = <ORG for original results or STD for standard results to derive the out of range values>  
    ,poptype    = <Y (for young population) or E (for elderly population)>  
    ,debug      = <Y (to keep temporary datasets) or N (to delete temporary datasets)>  
);
```

Macro to derive the PCSA

■ Two system datasets

■ PCSA Definitions



```
%st_saflags(  
  domain      = <domain: LB or EG or VS>  
  ,phase      = <phase: 1 (for phase 1) or 3 (for phase 2/3)>  
  ,dsin       = <input dataset>  
  ,dsout      = <output dataset>  
  ,dsoutpcsa  = <output dataset to store the PCSA definitions of the study parameters>  
  ,dspcsa     = <system dataset for PCSA definitions>  
  ,dsref      = <system dataset to link macro-variables and values>  
  ,cutoff     = <cut-off date to select version of the PCSA>  
  ,lnrefer    = <ORG for original results or STD for standard results to derive the out of range values>  
  ,poptype    = <Y (for young population) or E (for elderly population)>  
  ,debug      = <Y (to keep temporary datasets) or N (to delete temporary datasets)>  
);
```

System Dataset: PCSA Definitions

	A	B	C	D	E	F	G	H	I	J	K
1	_PCSADT	_TEST3	_TEST4	_OC1	_PCSAID	_PHASE	_DOMAIN	_PCSACL	_PCSAN	_PCSA	_PCSACOND
2	\$10	\$8	\$8	\$8	\$30	8	\$2	\$1	8	\$200	\$300
72	06/09/2000	CCRIT	CREAT	CREAT	CREAT_AD_+1	1	LB	+	1	>= 150 umol/L (Adults)	(&age= . or &age > 12) and upcase(&stunit) = 'UMOL/L' and &stresne . and
73	06/09/2000	CCRIT	CREAT	CREAT	CREAT_AD_0	1	LB		0	Normal	&age > 12 and upcase(&stunit) = 'UMOL/L' and &stresne . and
74	01/05/2007	CCRIT	CREAT	CREAT	CREAT_AD_0	1	LB		0		(&age= . or &age > 12) and upcase(&stunit) = 'UMOL/L' and &stresne . and
75	06/09/2000	CCRIT	CREAT	CREAT	CREAT_YG_+1	1	LB	+	1	>= 90 umol/L (6-12 year-old)	(6 <= &age <= 12) and upcase(&stunit) = 'UMOL/L' and &stresne . and
76	01/01/2007	CCRIT	CREAT	CREAT	CREAT_YG_0	1	LB		0	Normal	(6 <= &age <= 12) and upcase(&stunit) = 'UMOL/L' and &stresne . and
77	01/05/2007	CCRIT	CREAT	CREAT	CREAT_YG_0	1	LB		0		(6 <= &age <= 12) and upcase(&stunit) = 'UMOL/L' and &stresne . and
78	01/05/2007	CCRIT	CREAT	CREAT	CREAT_CHG_+1	1	LB	+	1	>= 30% from baseline	&chgbir >= 30
79	05/01/2009	CCRIT	CREAT	CREAT	CREAT_CHG_+1	1	LB	+	1	>= 30% and < 100% from baseline	30 <= &chgbir < 100
80	05/01/2009	CCRIT	CREAT	CREAT	CREAT_CHG_+2	1	LB	+	2	>= 100% from baseline	&chgbir >= 100
81	01/05/2007	CCRIT	CREAT	CREAT	CREAT_CHG_0	1	LB		0		&chgbir ne . and &chgbir < 30
82	01/05/2007	CRP	CRP	CRP	CRP_U1_+1	1	LB	+	1	> 2 ULN or > 10 mg/L (if ULN not provided)	&stnrhi= . and upcase(&stunit) = 'MG/L' and &stresne > 10
83	01/05/2007	CRP	CRP	CRP	CRP_U1_0	1	LB		0		&stnrhi= . and upcase(&stunit) = 'MG/L' and &stresne . and &
84	01/05/2007	CRP	CRP	CRP	CRP_U2_+1	1	LB	+	1	> 2 ULN or > 10 mg/L (if ULN not provided)	&stnrhi ne . and &stresne > (2 * &stnrhi)

%st_saflags(

domain = <domain: **LB** or **EG** or **VS**>

,phase = <phase: **1** (for phase 1) or **3** (for phase 2/3)>

,dsin = <input dataset>

,dsout = <output dataset>

,dsoutpcsa= <output dataset to store the PCSA definitions of the study parameters>

,dspcsa = <system dataset for PCSA definitions>

,dsref = <system dataset to link macro-variables and values>

,cutoff = <cut-off date to select version of the PCSA>

,lnrefer = <**ORG** for original results or **STD** for standard results to derive the out of range values>

,poptype = <**Y** (for young population) or **E** (for elderly population)>

,debug = <**Y** (to keep temporary datasets) or **N** (to delete temporary datasets)>

);

System Dataset: PCSA Definitions

	A	B	C	D	E	F	G	H	I	J	K
1	_PCSADT	_TEST3	_TEST4	_OC1	_PCSAID	_PHASE	_DOMAIN	_PCSACL	_PCSAN	_PCSA	_PCSACOND
2	\$10	\$8	\$8	\$8	\$30	8	\$2	\$1	8	\$200	\$300
72	06/09/2000	CCRIT	(&age=. or &age > 12) and upcase(&stunit) = 'UMOL/L' and &stresn >= 150								
73	06/09/2000	CCRIT	OL/L' and &stresn ne . and								
74	01/05/2007	CCRIT	CREAT	CREAT	CREAT_AD_0	1	LB		0		(&age=. or &age > 12) and upcase(&stunit) = 'UMOL/L' and &stresn
75	06/09/2000	CCRIT	CREAT	CREAT	CREAT_YG_+1	1	LB	+	1	>= 90 umol/L (6-12 year-old)	(6 <= &age <= 12) and upcase(&stunit) = 'UMOL/L' and &stresn
76	01/01/2007	CCRIT	CREAT	CREAT	CREAT_YG_0	1	LB		0	Normal	(6 <= &age <= 12) and upcase(&stunit) = 'UMOL/L' and &stresn
77	01/05/2007	CCRIT	CREAT	CREAT	CREAT_YG_0	1	LB		0		(6 <= &age <= 12) and upcase(&stunit) = 'UMOL/L' and &stresn
78	01/05/2007	CCRIT	CREAT	CREAT	CREAT_CHG_+1	1	LB	+	1	>= 30% from baseline	&chgbir >= 30
79	05/01/2009	CCRIT	30 <= &chgbir < 100				LB	+	1	>= 30% and < 100% from baseline	30 <= &chgbir < 100
80	05/01/2009	CCRIT					LB	+	2	>= 100% from baseline	&chgbir >= 100
81	01/05/2007	CCRIT	CREAT	CREAT	CREAT_CHG_0	1	LB		0		&chgbir ne . and &chgbir < 30
82	01/05/2007	CRP	CRP	CRP	CRP_U1_+1	1	LB	+	1	> 2 ULN or > 10 mg/L (if ULN not provided)	&stnrhi= . and upcase(&stunit) = 'MGL' and &stresn > 10
83	01/05/2007	CRP	&stnrhi ne . and &stresn > (2 * &stnrhi)						0		&stnrhi= . and upcase(&stunit) = 'MGL' and &stresn ne . and &
84	01/05/2007	CRP							1	> 2 ULN or > 10 mg/L (if ULN not provided)	&stnrhi ne . and &stresn > (2 * &stnrhi)

%st_saflags(

domain = <domain: **LB** or **EG** or **VS**>

,phase = <phase: **1** (for phase 1) or **3** (for phase 2/3)>

,dsin = <input dataset>

,dsout = <output dataset>

,dsoutpcsa= <output dataset to store the PCSA definitions of the study parameters>

,dspcsa = <system dataset for PCSA definitions>

,dsref = <system dataset to link macro-variables and values>

,cutoff = <cut-off date to select version of the PCSA>

,lnrefer = <**ORG** for original results or **STD** for standard results to derive the out of range values>

,poptype = <**Y** (for young population) or **E** (for elderly population)>

,debug = <**Y** (to keep temporary datasets) or **N** (to delete temporary datasets)>

);

Macro to derive the PCSA

■ Two system datasets

■ PCSA Definitions



■ Macro-variables used in the macro



```
%st_saflags(  
  domain      = <domain: LB or EG or VS>  
  ,phase      = <phase: 1 (for phase 1) or 3 (for phase 2/3)>  
  ,dsin       = <input dataset>  
  ,dsout      = <output dataset>  
  ,dsoutpcsa  = <output dataset to store the PCSA definitions of the study parameters>  
  ,dspcsa     = <system dataset for PCSA definitions>  
  ,dsref      = <system dataset to link macro-variables and values>  
  ,cutoff     = <cut-off date to select version of the PCSA>  
  ,lnrefer    = <ORG for original results or STD for standard results to derive the out of range values>  
  ,poptype    = <Y (for young population) or E (for elderly population)>  
  ,debug      = <Y (to keep temporary datasets) or N (to delete temporary datasets)>  
);
```

CALL SET Routine and FETCH Function

- SAS® definitions:

- **CALL SET Routine**

- Links SAS data set variables to DATA step or macro variables that have the same name and data type

- **FETCH Function**

- Reads the next nondeleted observation from a SAS data set into the Data Set Data Vector (DDV)

CALL SET Routine and FETCH Function

■ Part 1: Creation of macro-variables

```
%let ref_dsid=%sysfunc(open(_st_dsref(where=(vardomain="&domain."))));  
%syscall set(ref_dsid);  
%do %while(%sysfunc(fetch(&ref_dsid.))=0);  
    data _null_;  
        %let &macname.=&varname.;  
        %let l_&macname.=&varlabel.;  
    run;  
%end;  
%let ref_rc=%sysfunc(close(&ref_dsid.));
```

CALL SET Routine and FETCH Function

```
%let &macname.=&varname.;
%let l_&macname.=&varlabel.;
```

	A	B	C	D	E	F	G
1	macname	dssource	vardomain	varname	varlabel	vartype	varstatus
2	\$10	\$10	\$2	\$20	\$200	\$3	\$6
3	sex	DSIN	LB	SEX	Sex	C	INPUT
4	race	DSIN	LB	RACE	Race	C	INPUT
5	age	DSIN	LB	AGE	Age	N	INPUT
6	testcd	DSIN	LB	LBTESTCD	Lab test	C	INPUT
7	orresn	DSIN	LB	LBORRESN	Numeric result (original units)	N	INPUT
8	ornrhi	DSIN	LB	LBORNRI	Reference range upper limit org units	N	INPUT
9	ornrlo	DSIN	LB	LBORNRL	Reference range lower limit org units	N	INPUT
10	stunit	DSIN	LB	LBSTRESU	Standard units	C	INPUT
11	stresn	DSIN	LB	LBSTRESN	Numeric result (standard units)	N	INPUT
12	stnr	%let sex=SEX;			Reference range upper limit std units	N	INPUT
13	stnr	%let l_sex=Sex;			Reference range lower limit std units	N	INPUT
14	fasti				Fasting status	C	INPUT
15	flag	%let race=RACE;			Baseline flag	C	INPUT
16	chgl	%let l_race=Race;			Change from baseline	N	INPUT
17	chgl				Relative change from baseline	N	INPUT
18	blpc	%let age=AGE;			Baseline status for PCSA analysis	C	OUTPUT
19	pcs:	%let l_age=Age;			PCSA criteria	C	OUTPUT
20	pcs:				PCSA identifier	C	OUTPUT
21	pcs:	%let testcd=LBTESTCD;			PCSA severity	N	OUTPUT
22	pcs:	%let l_testcd=Lab test;			PCSA criteria for change	C	OUTPUT
23	pcs:				PCSA identifier for change	C	OUTPUT
24	pcs:	%let orresn=LBORRESN;			PCSA severity for change	N	OUTPUT
25	nrin	%let l_orresn=Numeric result (original units);			Reference indicator range	C	OUTPUT
26	age	DSIN	VS	AGE	Age	N	INPUT
27	testcd	DSIN	VS	VSTESTCD	Vital signs test	C	INPUT
28	stunit	DSIN	VS	VSSTRESU	Standard units	C	INPUT
29	stresn	DSIN	VS	VSSTRESN	Numeric result	N	INPUT

CALL SET Routine and FETCH Function

■ Part 2: Resolution of the macro-variables

```
%let id_pcsa=%sysfunc(open(_st_pcsa,i));
%if &id_pcsa. gt 0 %then %do;
  %syscall set(id_pcsa);
  data _st_temp;
    length pcsa $200 pcsafl $1 pcsan 8 pcsaid $30;
    set _st_dsin;
    %do %while(%sysfunc(fetch(&id_pcsa.))=0);
      if trim(&testcd.)="&_oc1." then do;
        if &_pcsacond. then do;
          pcsa =trim("&_pcsa.");
          pcsafl =trim("&_pcsafl.");
          %if &_pcsan. ne . %then %str(pcsan =%eval(&_pcsan.););
          %else %str(pcsan=.););
          pcsaid=trim("&_pcsaid.");
          output;
        end;
      end;
    %end;
  run;
%end;
%let rc_pcsa=%sysfunc(close(&id_pcsa.));
```

CALL SET Routine and FETCH Function

```
%do %while(%sysfunc(fetch(&id_pcsa.))=0);
  if trim(&testcd.)="_oc1." then do;
    if &_pcsacond. then do;
      pcsa =trim("&_pcsa.");
      pcsafl =trim("&_pcsafl.");
      %if &_pcsan. ne . %then %str(pcsan =%eval(&_pcsan.););
      %else %str(pcsan=.););
      pcsaid=trim("&_pcsaid.");
      output;
    end;
  %end;
%end;
```

	A	B	C	D	E	F	G	H	I	J	K
	_PCSADT	_TEST	_TEST	_OC1	_PCSAID	_PHASE	_DOMAIN	_PCSAFL	_PCSAN	_PCSA	_PCSACOND
5	05/01/2009	CALKPH	ALP	ALP	ALP +1	1	LB	+	1	> 1.5 ULN	&stnrhi ne . and &stresn > (1.5 * &stnrhi)
8	05/01/2009	CALKPH	ALP	ALP	ALP_0	1	LB		0		&stnrhi ne . and &stresn ne . and &stresn <= (1.5 * &stnrhi)
11	05/01/2009	CALT	ALT	ALT	ALT +1	1	LB	+	1	> 3 ULN and <= 5 ULN	&stnrhi ne . and (3 * &stnrhi) < &stresn <= (5 * &stnrhi)
14	05/01/2009	CALT	ALT								5 * &stnrhi) < &stresn <= (10 * &stnrhi)
17	05/01/2009	CALT	ALT								10 * &stnrhi) < &stresn <= (20 * &stnrhi)
20	05/01/2009	CALT	ALT								&stresn > (20 * &stnrhi)
25	05/01/2009	CALT	ALT								&stresn ne . and &stresn <= (3 * &stnrhi)
28	05/01/2009	CAST	AST								3 * &stnrhi) < &stresn <= (5 * &stnrhi)
31	05/01/2009	CAST	AST								5 * &stnrhi) < &stresn <= (10 * &stnrhi)
34	05/01/2009	CAST	AST								10 * &stnrhi) < &stresn <= (20 * &stnrhi)
37	05/01/2009	CAST	AST								&stresn > (20 * &stnrhi)
42	05/01/2009	CAST	AST								&stresn ne . and &stresn <= (3 * &stnrhi)
51	05/01/2009	CCK	CPK								3 * &stnrhi) < &stresn <= (10 * &stnrhi)
52	05/01/2009	CCK	CPK								&stresn > (10 * &stnrhi)
55	05/01/2009	CCK	CPK								&stresn ne . and &stresn <= (3 * &stnrhi)
79	05/01/2009	CCRIT	CREAT								00
80	05/01/2009	CCRIT	CREAT								

```
if trim(&testcd.)="ALP" then do;
  if &stnrhi ne . and &stresn > (1.5 * &stnrhi) then do;
    pcsa =trim("> 1.5 ULN");
    pcsafl =trim("+");
    pcsan =1;
    pcsaid=trim("ALP_+1");
    output;
  end;
end;
if trim(&testcd.)="ALP" then do;
  if &stnrhi ne . and &stresn ne . and &stresn <= (1.5 * &stnrhi) then do;
    pcsa =trim("");
    pcsafl =trim("");
    pcsan =0;
    pcsaid=trim("ALP_0");
    output;
  end;
end;
```

Conclusion

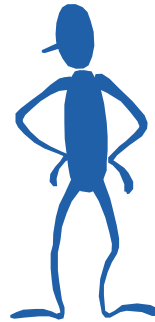
- The standardization of the PCSA definitions and the automation of their computation with a unique macro ensure an homogeneity of the PCSA analysis through all the therapeutic areas world-wide
- The CALL SET routine and the function FETCH allow to manage an important number of macro-variables and SAS® conditions/instructions thanks to system datasets...
- ... and these system datasets symplify also the updates of the PCSA definitions

To finish ...

- Thank you to all the persons who have worked or who work always on this macro:
 - Fabienne Rouillé
 - Yvane Boudraa
- And especially
 - Gérard Capdevielle who is the father of this macro

Thank you for your attention ...
... and see you next year !!

Questions ? ? ?



Contact:

claudio.guyot@sanofi-aventis.com

(33) 1 69 79 72 72