SDTM-ETL 4.x User Manual and Tutorial

Author: Jozef Aerts, XML4Pharma

Last update: 2022-06-20

Working with ValueLists and the WhereClause in define.xml

As of define.xml 2.0, it is mandatory to define under which circumstances **value level metadata** is used, using so-called „WhereClauses“.
For example, for VSORRES, the properties of the value itself will differ depending on what the value of VSTESTCD is. For example, when VSTESTCD=DIABP (diastolic blood pressure), we expect the value to be an integer between 80 and 120 (mm[Hg]) or so. In case VSTESTCD=WEIGHT, we probably expect a floating point number, and the unit either being kg or pounds. For some other tests, we might however expect a string, like "S“, "M", "L" or "XL" in the case that the test is "FRMSIZE" (frame size).

In this short tutorial, we will describe a somewhat more complicated example which is also described in the define.xml 2.0 specification: we will describe that the weight of a subject is described in pounds in case the country of the subject is the USA, and in kilograms in case the country of the subject is France or Germany[1]. Such a condition is described in define.xml 2.0 using a so-called "WhereClause".

"WhereClause" did not exist in define.xml 1.0. Instead, some people used nested "ValueLists" instead. In define.xml 2.0 and higher, nested Valuelists are not necessary anymore and should not be used.

Now, important is that we assign the "WhereClauses" to the correct SDTM variable. Because in the mentioned example the information is about units used, the "WhereClauses" needs to be assigned to VSORRESU.
In case that we state that mm[Hg] is an integer for VSTESTCD=SYSBP and DIABP, and that the value is of type "text" enumerated to "S", "M", "L" and "XL" for VSTESTCD="FRMSIZE" the "WhereClauses" needs to be assigned to VSORRES.

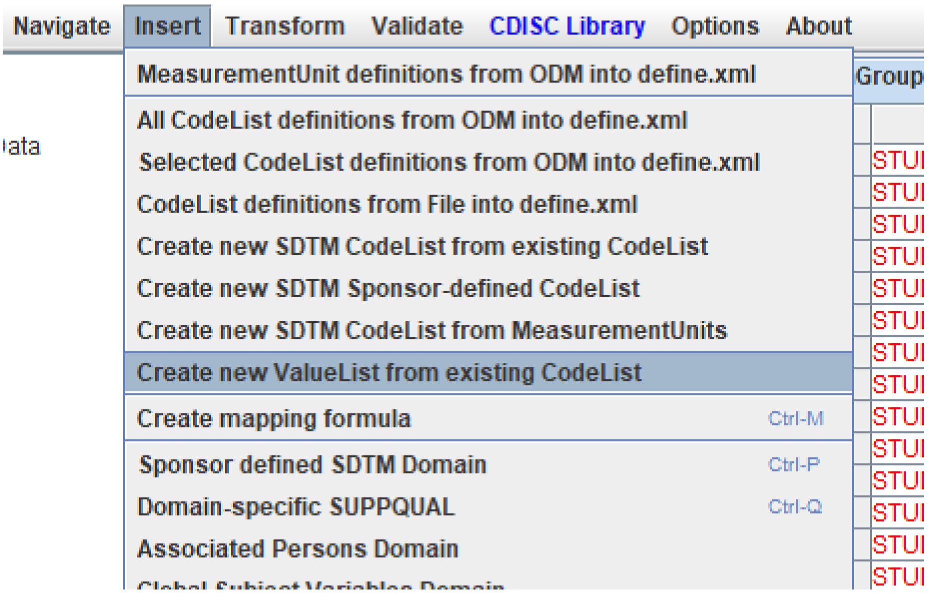
So what we need to express is:

- the value of VSORRES is an integer in case VSTESTCD=DIABP or SYSBP
- the value of VSORRES is of type text and enumerated to "S", "M", "L" and "XL" when VSTESTCD="FRMSIZE"
- the unit in VSORRESU is "pounds" when VSTESTCD=WEIGHT and DM.COUNTRY=USA
- the unit in VSORRESU is "kilogram" when VSTESTC=WEIGHT and DM.COUNTRY=GER or DM.COUNTRY=FRA

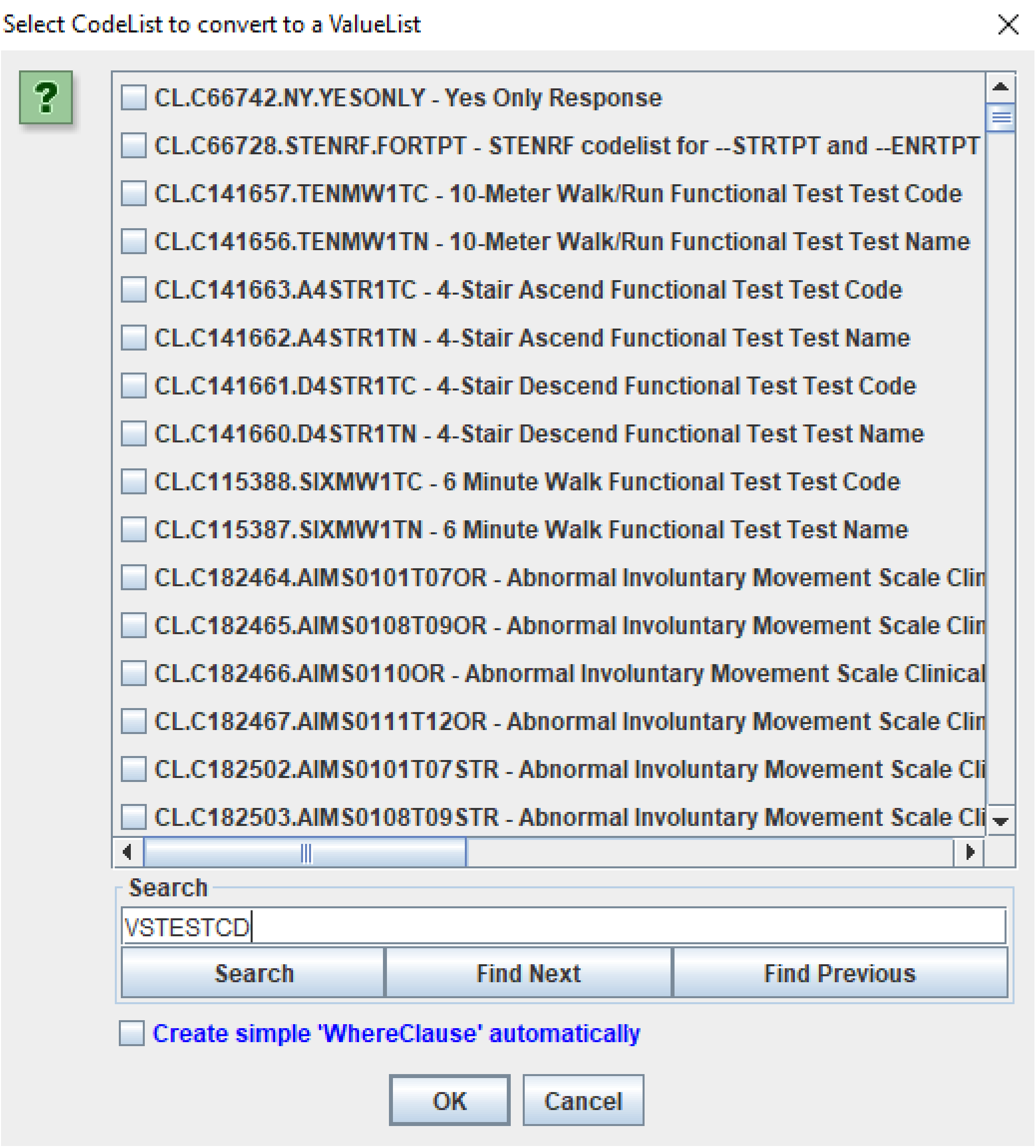
[1] In the define.xml 2.0 specification, Mexico and Canada are mentioned for the use of kilograms, but using France and Germany is more clear especially for non-US users.

Creating a ValueList for VSORRES

Let us first create a valuelist for VSORRES. The easiest is to start from an existing codelist, and use the menu "Insert – Create New ValueList from existing CodeList":


Navigate
Insert
Transform
Validate
CDISC Library
Options
About
MeasurementUnit definitions from ODM into define.xml
All CodeList definitions from ODM into define.xml
Selected CodeList definitions from ODM into define.xml
CodeList definitions from File into define.xml
Create new SDTM CodeList from existing CodeList
Create new SDTM Sponsor-defined CodeList
Create new SDTM CodeList from MeasurementUnits
Create new ValueList from existing CodeList
Create mapping formula Ctrl-M
Sponsor defined SDTM Domain Ctrl-P
Domain-specific SUPPQUAL Ctrl-Q
Associated Persons Domain
Group
ata
STU

and look for "VSTESTCD",



Select CodeList to convert to a ValueList

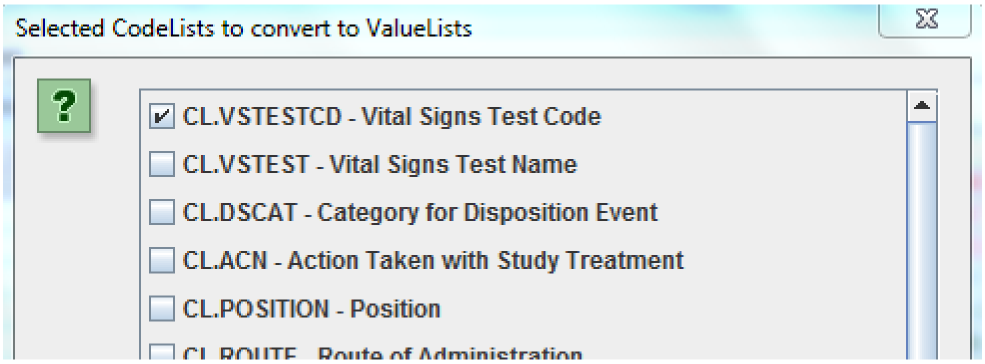
- □ CL.C66742.NY.YESONLY - Yes Only Response
- □ CL.C66728.STENRF.FORTPT - STENRF codelist for --STRTPT and --ENRTPT
- □ CL.C141657.TENMW1TC - 10-Meter Walk/Run Functional Test Test Code
- □ CL.C141656.TENMW1TN - 10-Meter Walk/Run Functional Test Test Name
- □ CL.C141663.A4STR1TC - 4-Stair Ascend Functional Test Test Code
- □ CL.C141662.A4STR1TN - 4-Stair Ascend Functional Test Test Name
- □ CL.C141661.D4STR1TC - 4-Stair Descend Functional Test Test Code
- □ CL.C141660.D4STR1TN - 4-Stair Descend Functional Test Test Name
- □ CL.C115388.SIXMW1TC - 6 Minute Walk Functional Test Test Code
- □ CL.C115387.SIXMW1TN - 6 Minute Walk Functional Test Test Name
- □ CL.C182464.AIMS0101T07OR - Abnormal Involuntary Movement Scale Clin
- □ CL.C182465.AIMS0108T09OR - Abnormal Involuntary Movement Scale Clin
- □ CL.C182466.AIMS0110OR - Abnormal Involuntary Movement Scale Clinical
- □ CL.C182467.AIMS0111T12OR - Abnormal Involuntary Movement Scale Clin
- □ CL.C182502.AIMS0101T07STR - Abnormal Involuntary Movement Scale Cli
- □ CL.C182503.AIMS0108T09STR - Abnormal Involuntary Movement Scale Cli

Search

VSTESTCD

□ Create simple 'WhereClause' automatically

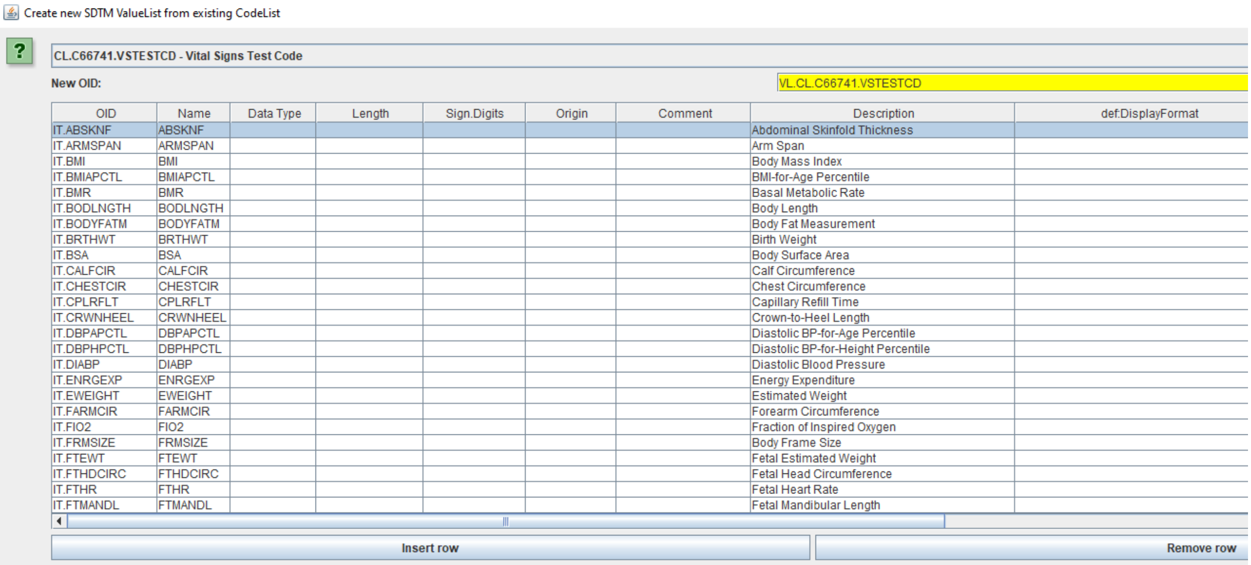
In this case, we will not check "Create simple 'WhereClause' automatically, as we want to develop a more complicated "where".
This is leading to:



Selected CodeLists to convert to ValueLists

- ☑ CL.VSTESTCD - Vital Signs Test Code
- □ CL.VSTEST - Vital Signs Test Name
- □ CL.DSCAT - Category for Disposition Event
- □ CL.ACN - Action Taken with Study Treatment
- □ CL.POSITION - Position
- □ CL.ROUTE - Route of Administration

where we start from CL.VSTESTCD as our metadata will depend on the value of VSTESTCD. This leads to a new dialog with:



Create new SDTM ValueList from existing CodeList

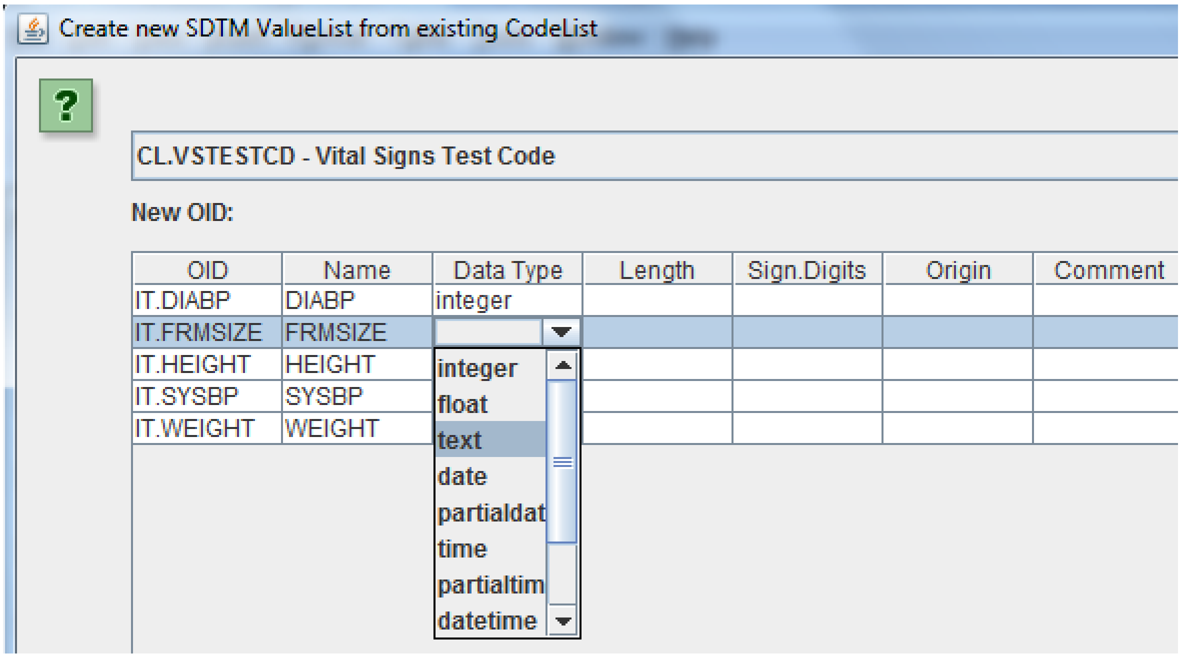
CL.C66741.VSTESTCD - Vital Signs Test Code

New OID: VL.CL.C66741.VSTESTCD

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:DisplayFormat
IT.ABSKNF	ABSKNF						Abdominal Skinfold Thickness	
IT.ARMSPAN	ARMSPAN						Arm Span	
IT.BMI	BMI						Body Mass Index	
IT.BMIAPCTL	BMIAPCTL						BMI-for-Age Percentile	
IT.BMR	BMR						Basal Metabolic Rate	
IT.BODLNGTH	BODLNGTH						Body Length	
IT.BODYFATM	BODYFATM						Body Fat Measurement	
IT.BRTHWT	BRTHWT						Birth Weight	
IT.BSA	BSA						Body Surface Area	
IT.CALFCIR	CALFCIR						Calf Circumference	
IT.CHESTCIR	CHESTCIR						Chest Circumference	
IT.CPLRFLT	CPLRFLT						Capillary Refill Time	
IT.CRWNHEEL	CRWNHEEL						Crown-to-Heel Length	
IT.DBPAPCTL	DBPAPCTL						Diastolic BP-for-Age Percentile	
IT.DBPHPCTL	DBPHPCTL						Diastolic BP-for-Height Percentile	
IT.DIABP	DIABP						Diastolic Blood Pressure	
IT.ENRGEXP	ENRGEXP						Energy Expenditure	
IT.EWEIGHT	EWEIGHT						Estimated Weight	
IT.FARMCIR	FARMCIR						Forearm Circumference	
IT.FIO2	FIO2						Fraction of Inspired Oxygen	
IT.FRMSIZE	FRMSIZE						Body Frame Size	
IT.FTEWT	FTEWT						Fetal Estimated Weight	
IT.FTHDCIRC	FTHDCIRC						Fetal Head Circumference	
IT.FTHR	FTHR						Fetal Heart Rate	
IT.FTMANDL	FTMANDL						Fetal Mandibular Length	

allowing us to assign metadata for each type of test. In this tutorial, we will limit ourselves to a few tests only, i.e. DIABP", "SYSBP", "HEIGHT", "WEIGHT" and "FRMSIZE", so we can remove all other rows. We will however only show the "WhereClause" for each of them, as they are all very similar, but with different values for the "check value" (see further).

After having removed the unnessary[2] rows(using the "Remove row" button), we can start adding the data types:



Create new SDTM ValueList from existing CodeList

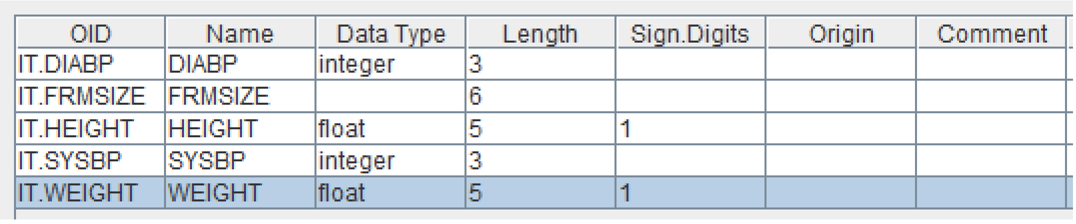
CL.VSTESTCD - Vital Signs Test Code

New OID:

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment
IT.DIABP	DIABP	integer				
IT.FRMSIZE	FRMSIZE					
IT.HEIGHT	HEIGHT					
IT.SYSBP	SYSBP					
IT.WEIGHT	WEIGHT					

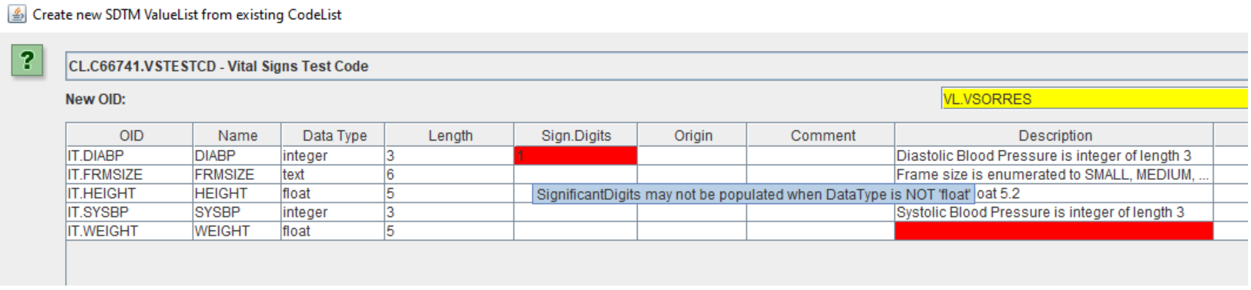
and appropriate maximal lengths and number of digits after the decimal point ("Significant Digits"):

[2] This depends of course on your own study.



OID	Name	Data Type	Length	Sign.Digits	Origin	Comment
IT.DIABP	DIABP	integer	3			
IT.FRMSIZE	FRMSIZE		6			
IT.HEIGHT	HEIGHT	float	5	1		
IT.SYSBP	SYSBP	integer	3			
IT.WEIGHT	WEIGHT	float	5	1		

One can always click the "Validate" button to check whether the combinations of filled fields makes sense.



Create new SDTM ValueList from existing CodeList

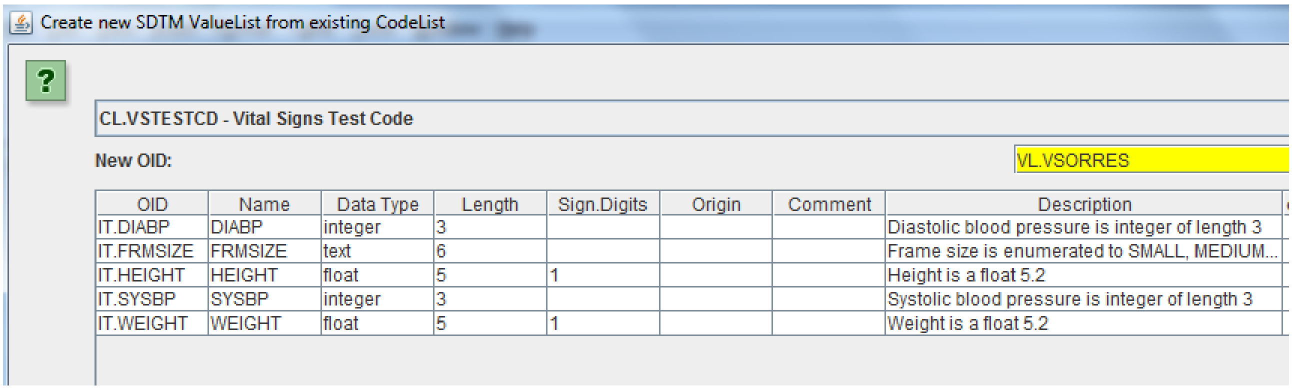
CL.C66741.VSTESTCD - Vital Signs Test Code

New OID: VL.VSORRES

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description
IT.DIABP	DIABP	integer	3	1			Diastolic Blood Pressure is integer of length 3
IT.FRMSIZE	FRMSIZE	text	6				Frame size is enumerated to SMALL, MEDIUM, ...
IT.HEIGHT	HEIGHT	float	5				oat 5.2
IT.SYSBP	SYSBP	integer	3				Systolic Blood Pressure is integer of length 3
IT.WEIGHT	WEIGHT	float	5				🟥

SignificantDigits may not be populated when DataType is NOT 'float'

Also notice that we changed the OID into "VL.VSORRES" to reflect that this is a valuelist for the variable VSORRES. We also see that the last "Description"[3] cell is colored red, meaning we do need provide a value there:



Create new SDTM ValueList from existing CodeList

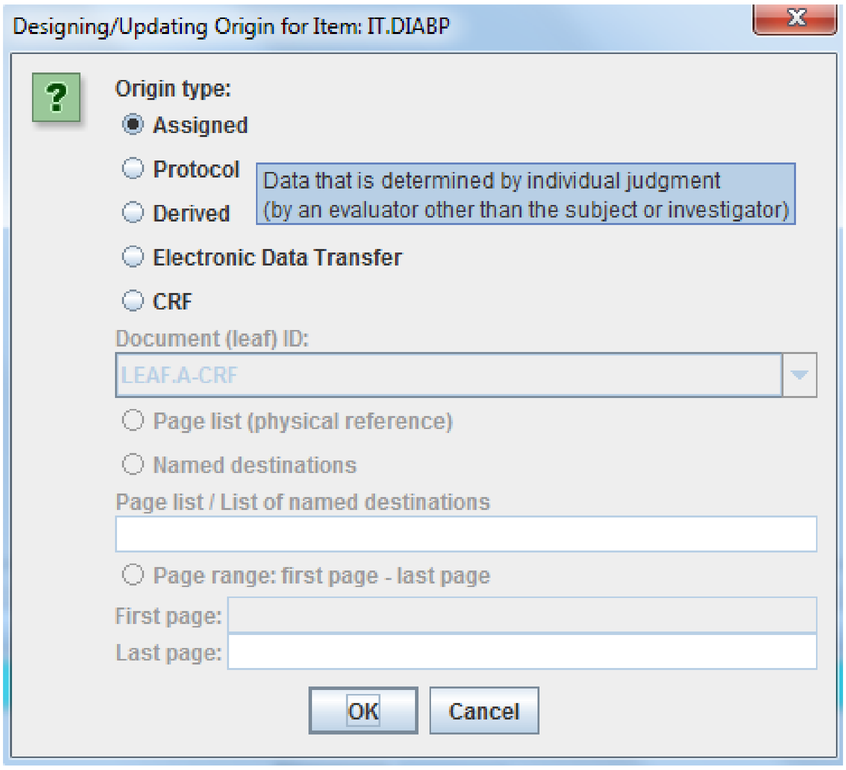
CL.VSTESTCD - Vital Signs Test Code

New OID: VL.VSORRES

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description
IT.DIABP	DIABP	integer	3				Diastolic blood pressure is integer of length 3
IT.FRMSIZE	FRMSIZE	text	6				Frame size is enumerated to SMALL, MEDIUM...
IT.HEIGHT	HEIGHT	float	5	1			Height is a float 5.2
IT.SYSBP	SYSBP	integer	3				Systolic blood pressure is integer of length 3
IT.WEIGHT	WEIGHT	float	5	1			Weight is a float 5.2

Now, we do also want to provide the reviewer the information what the origin of the data is. This can of course depend of the test code. In our case, we had a CRF with all vital signs on one page. So when clicking the cell "Origin" for "DIABP" the following dialog is displayed:

[3] "Description" replaces "def:Label" in define.xml v.2.0



Designing/Updating Origin for Item: IT.DIABP

Origin type:

● Assigned

○ Protocol

○ Derived

Data that is determined by individual judgment
(by an evaluator other than the subject or investigator)

○ Electronic Data Transfer

○ CRF

Document (leaf) ID: LEAF.A-CRF

○ Page list (physical reference)

○ Named destinations

Page list / List of named destinations:

○ Page range: first page - last page

First page:

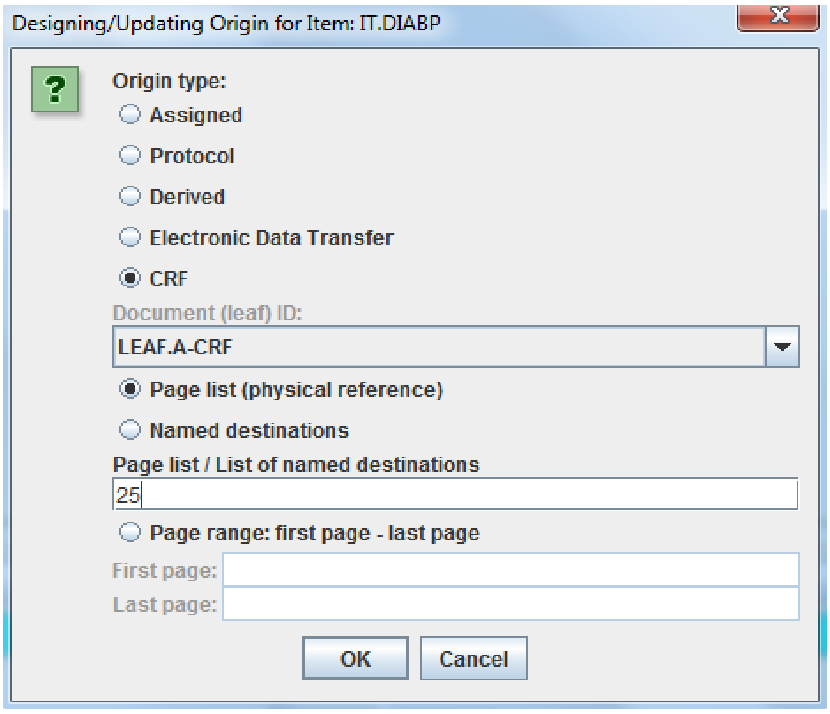
Last page:

where can select between[4]:

- Assigned: judgement from evaluator not being the investigator
- Protocol: prescribed by the protocol
- Derived: calculated using some algorithm
- Electronic Data Transfer (eDT): e.g. ECG data, lab data
- CRF: case report form

In our case, we of course select "CRF". Some other fields become available:

[4] In the case of Define-XML 2.1, the dialog will look a bit different.



Designing/Updating Origin for Item: IT.DIABP

Origin type:

○ Assigned

○ Protocol

○ Derived

○ Electronic Data Transfer

● CRF

Document (leaf) ID: LEAF.A-CRF

● Page list (physical reference)

○ Named destinations

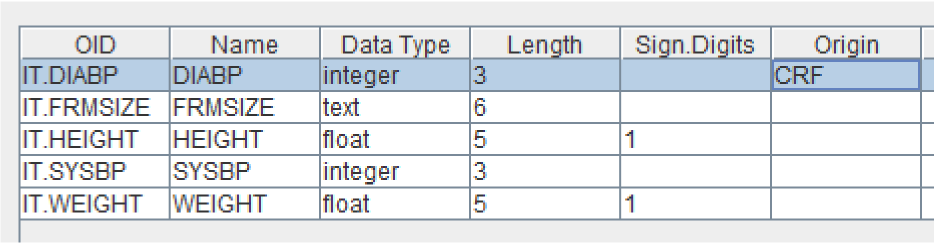
Page list / List of named destinations: 25

○ Page range: first page - last page

First page:

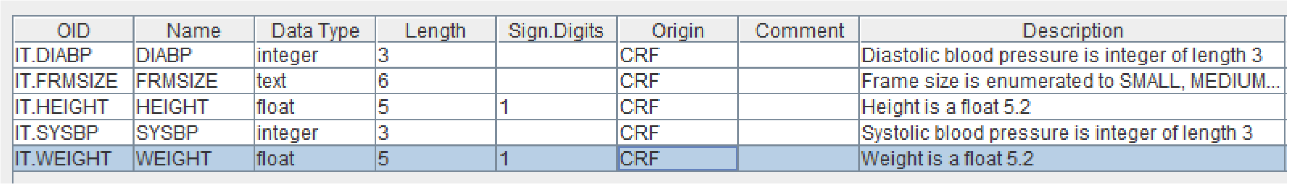
Last page:

We can either choose between a single page or page list (e.g.: 21 24 27) or a PDF named destination or a page range where we then need to provide the first and last page. In our case, the field that goes into VSORRES for VSTESTCD=DIABP can be found on page 25 of the annotated CRF. Clicking OK leads to:



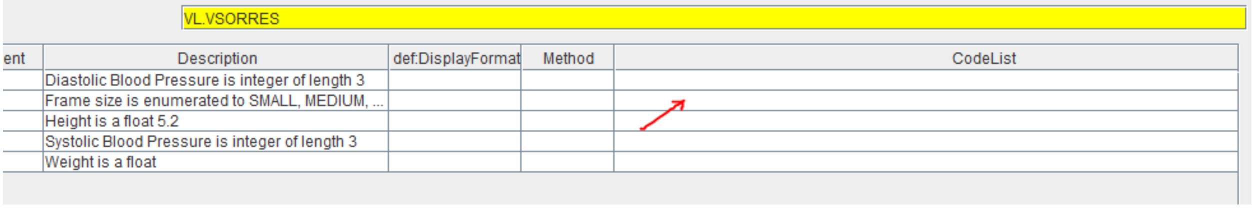
OID	Name	Data Type	Length	Sign.Digits	Origin
IT.DIABP	DIABP	integer	3		CRF
IT.FRMSIZE	FRMSIZE	text	6		
IT.HEIGHT	HEIGHT	float	5	1	
IT.SYSBP	SYSBP	integer	3		
IT.WEIGHT	WEIGHT	float	5	1	

and we can now to the same for the other rows, leading to:



OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description
IT.DIABP	DIABP	integer	3		CRF		Diastolic blood pressure is integer of length 3
IT.FRMSIZE	FRMSIZE	text	6		CRF		Frame size is enumerated to SMALL, MEDIUM...
IT.HEIGHT	HEIGHT	float	5	1	CRF		Height is a float 5.2
IT.SYSBP	SYSBP	integer	3		CRF		Systolic blood pressure is integer of length 3
IT.WEIGHT	WEIGHT	float	5	1	CRF		Weight is a float 5.2

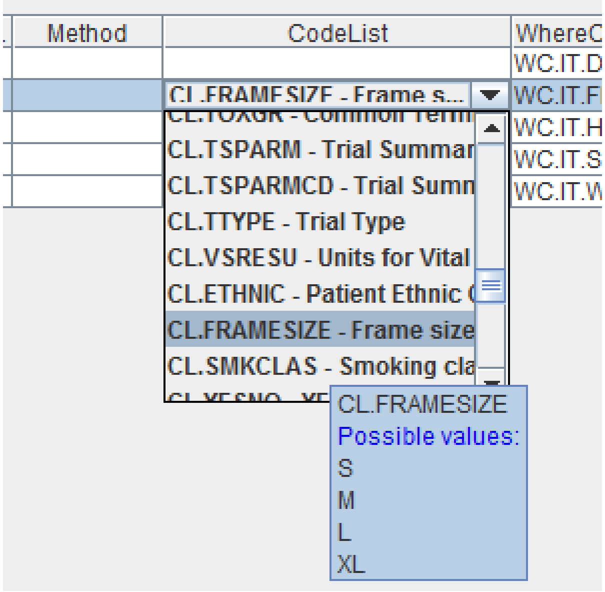
For "FRMSIZE" we need to indicate that the value is enumerated. This is usually done by a CodeList, so we click the cell "CodeList" for row "FRMSIZE",



VL.VSORRES

ent	Description	def:DisplayFormat	Method	CodeList
	Diastolic Blood Pressure is integer of length 3			
	Frame size is enumerated to SMALL, MEDIUM, ...			
	Height is a float 5.2			
	Systolic Blood Pressure is integer of length 3			
	Weight is a float			

leading to:



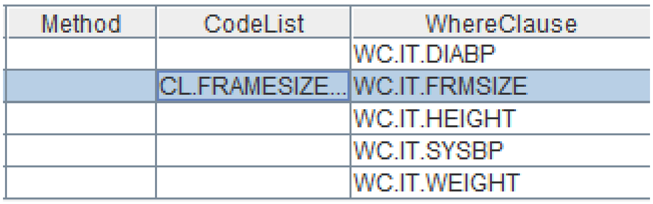
Method	CodeList	WhereC
		WC.IT.D
	CL.FRAMESIZE - Frame s...	WC.IT.F
		WC.IT.H
		WC.IT.S
		WC.IT.W

CL.TOXGR - Common Term
CL.TSPARM - Trial Summar
CL.TSPARMCD - Trial Summ
CL.TTYPE - Trial Type
CL.VSRESU - Units for Vital
CL.ETHNIC - Patient Ethnic
CL.FRAMESIZE - Frame size
CL.SMKCLAS - Smoking cla
CL.YESNO - YE

CL.FRAMESIZE
Possible values:
S
M
L
XL

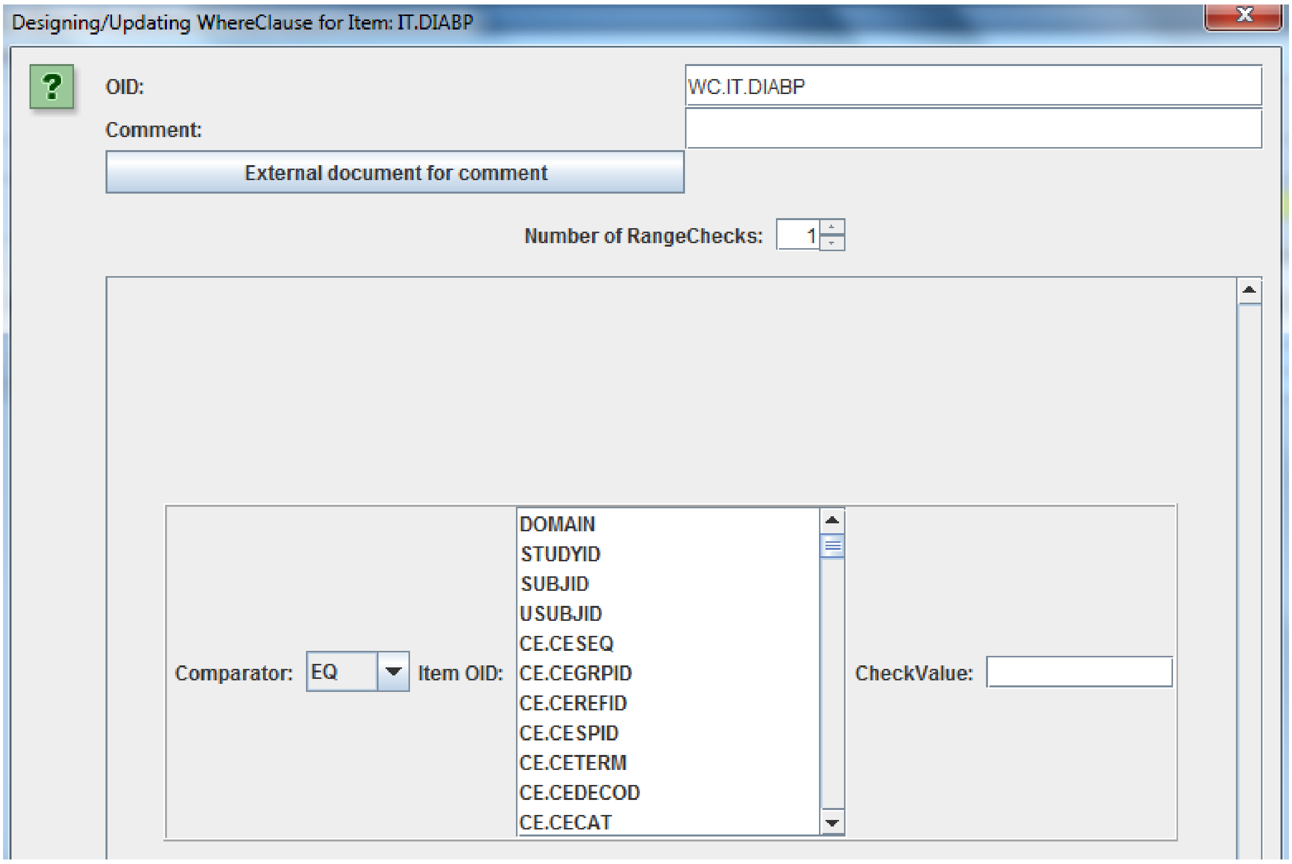
and scroll down, or use the "Search" button until we find the appropriate codelist. Notice that the allowed values are displayed in a tooltip, making it easier to find the correct one.

All we still need to do (mandatory) is to provide the "WhereClause" for each row, indicating under which conditions the metadata (data type, length, enumeration) need to be used:



Method	CodeList	WhereClause
		WC.IT.DIABP
	CL.FRAMESIZE...	WC.IT.FRMSIZE
		WC.IT.HEIGHT
		WC.IT.SYSBP
		WC.IT.WEIGHT

In the "WhereClauses", we will need to define that the metadata for "DIABP" need to be used when VSTESTCD=DIABP (simple isn't it?). So clicking the cell WC.IT.DIABP (the system already provides a proposal for the identifier), the following dialog is displayed:



Designing/Updating WhereClause for Item: IT.DIABP

OID: WC.IT.DIABP

Comment:

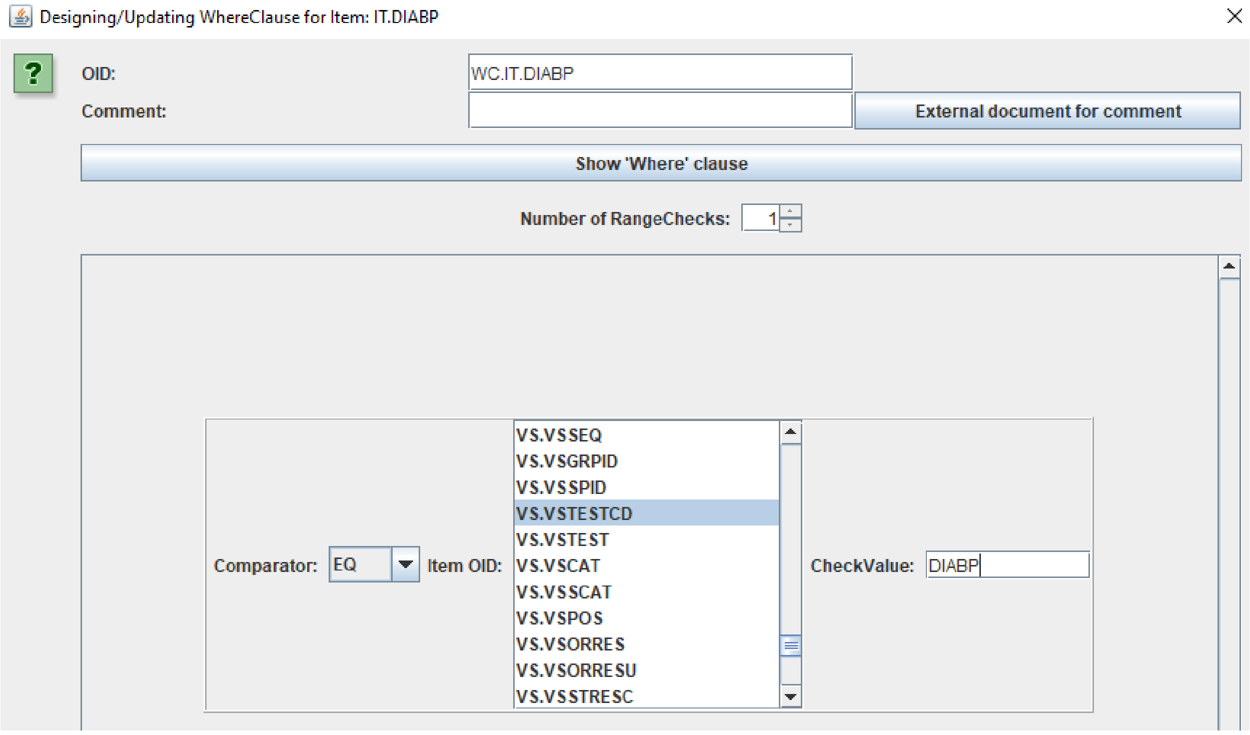
Number of RangeChecks: 1

Comparator: EQ

Item OID:

CheckValue:

The only thing we need to state as condition is that: VSTESTCD=DIABP, which easily accomplished by:



Designing/Updating WhereClause for Item: IT.DIABP

OID: WC.IT.DIABP

Comment:

Number of RangeChecks: 1

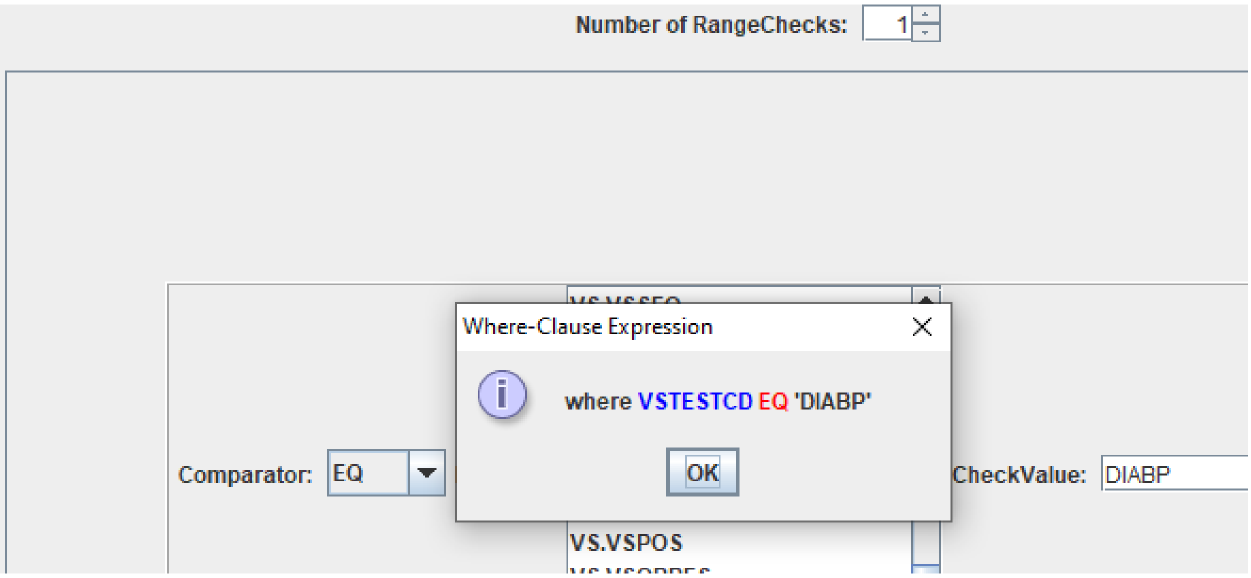
Comparator: EQ

Item OID: VS.VSTESTCD

CheckValue: DIABP

using the comparator "EQ" which means "equals to", which is exactly what we want.

If in doubt, one can use the button "Show 'Where' clause", providing:


Number of RangeChecks: 1

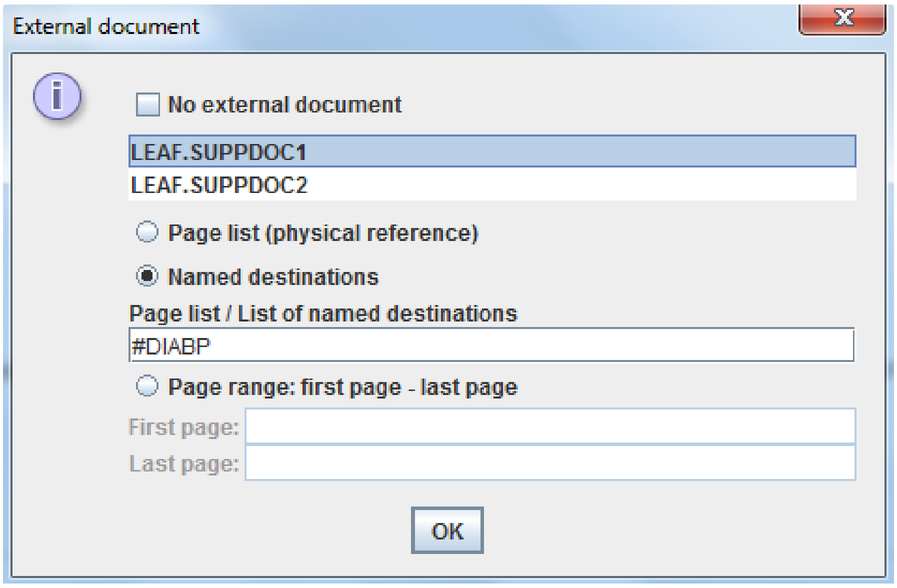
Where-Clause Expression

where VSTESTCD EQ 'DIABP'

Comparator: EQ

CheckValue: DIABP

We can also add a comment, and for the comment, even a link to an external document. Adding a comment is especially important when the condition is complex (see further) so that one can a textual declaration of the condition. If the condition is very complex, one may also provide a link to a specific page or named destination of an external document by clicking the "External document for comment" button, leading to:


External document

☐ No external document

LEAF.SUPPDOC1
LEAF.SUPPDOC2

○ Page list (physical reference)

● Named destinations

Page list / List of named destinations: #DIABP

○ Page range: first page - last page

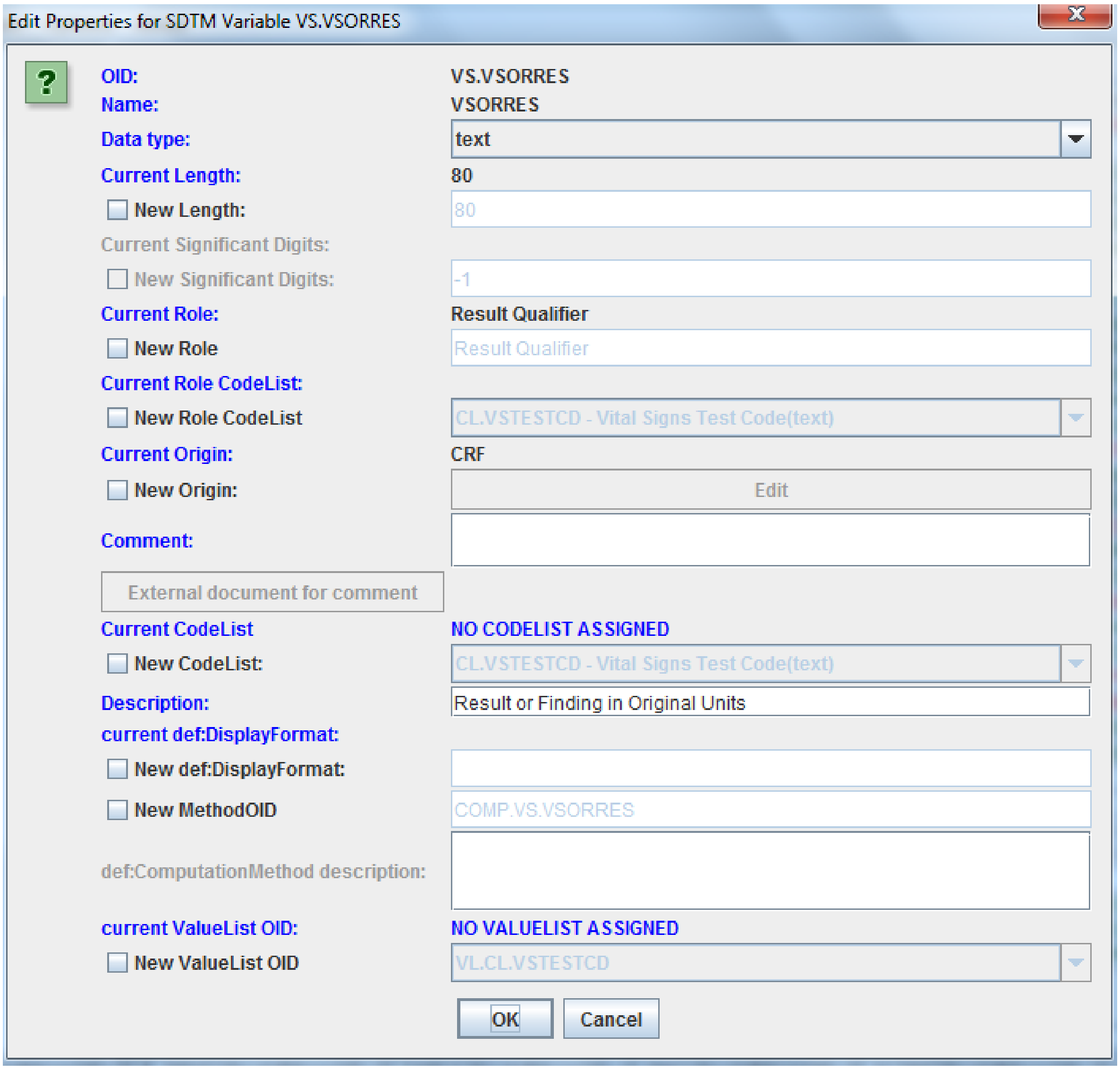
First page:

Last page:

in this case to a named destination within a PDF document.

Of course we also need to create the "WhereClause" for each other row, and adapt the condition accordingly (for example for "WEIGHT", we must state VSTESTCD=WEIGHT).

One thing we should NOT forget, is to assign the new ValueList with all its "WhereClauses" to the SDTM variable "VSORRES". This is done by selecting VSORRES in the main table and then using the menu "Edit – SDTM Variable Properties":



Edit Properties for SDTM Variable VS.VSORRES

OID: VS.VSORRES

Name: VSORRES

Data type: text

Current Length: 80

□ New Length: 80

Current Significant Digits:

□ New Significant Digits: -1

Current Role: Result Qualifier

□ New Role: Result Qualifier

Current Role CodeList:

□ New Role CodeList: CL.VSTESTCD - Vital Signs Test Code(text)

Current Origin: CRF

□ New Origin: Edit

Comment:

External document for comment

Current CodeList: NO CODELIST ASSIGNED

□ New CodeList: CL.VSTESTCD - Vital Signs Test Code(text)

Description: Result or Finding in Original Units

current def:DisplayFormat:

□ New def:DisplayFormat:

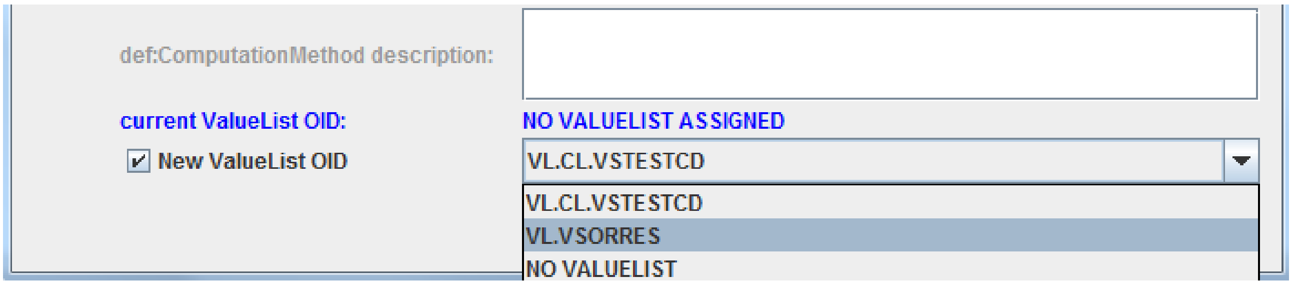
□ New MethodOID: COMP.VS.VSORRES

def:ComputationMethod description:

current ValueList OID: NO VALUELIST ASSIGNED

□ New ValueList OID: VL.CL.VSTESTCD

and then use the checkbox "New ValueList OID", and selecting the newly created ValueList "VL.VSORRES"

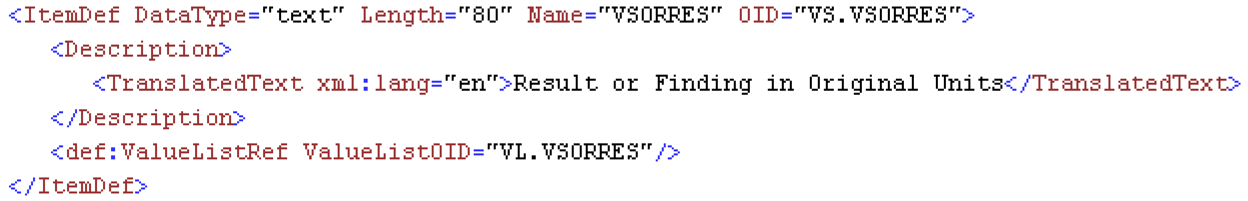


def:ComputationMethod description:

current ValueList OID: NO VALUELIST ASSIGNED

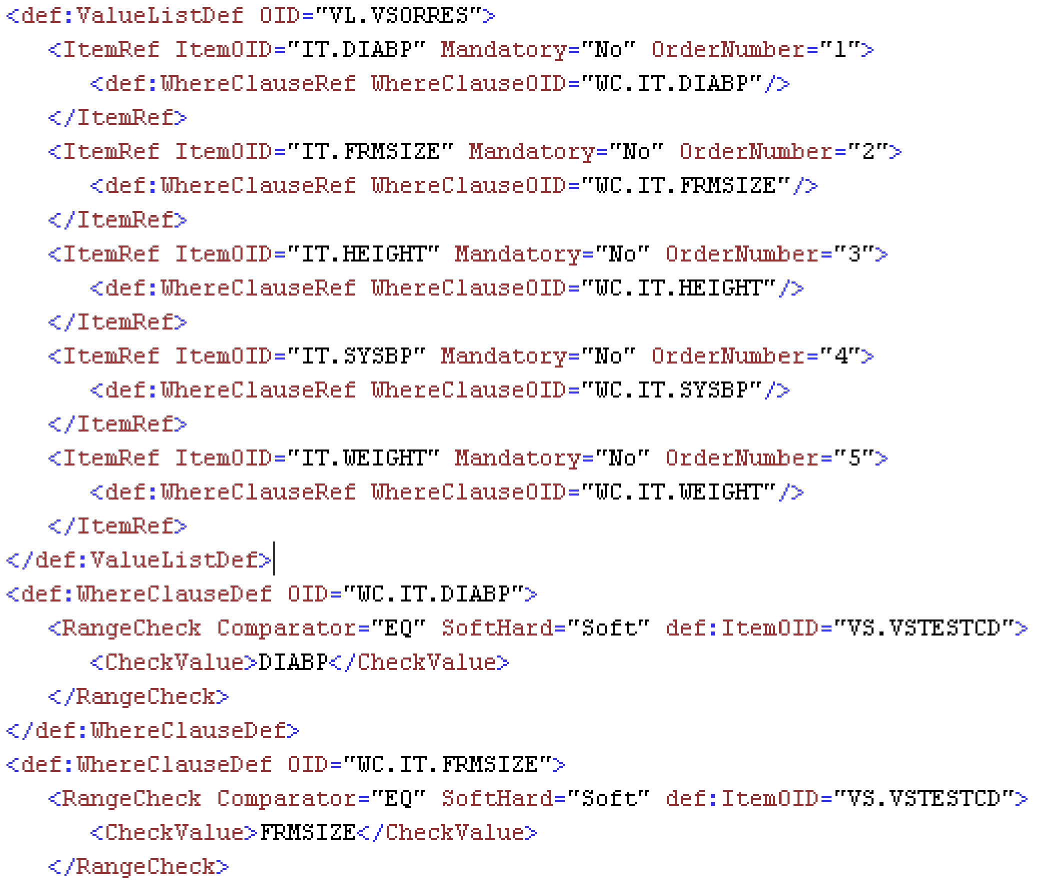
☑ New ValueList OID: VL.CL.VSTESTCD

In the define.xml, this will later look like:



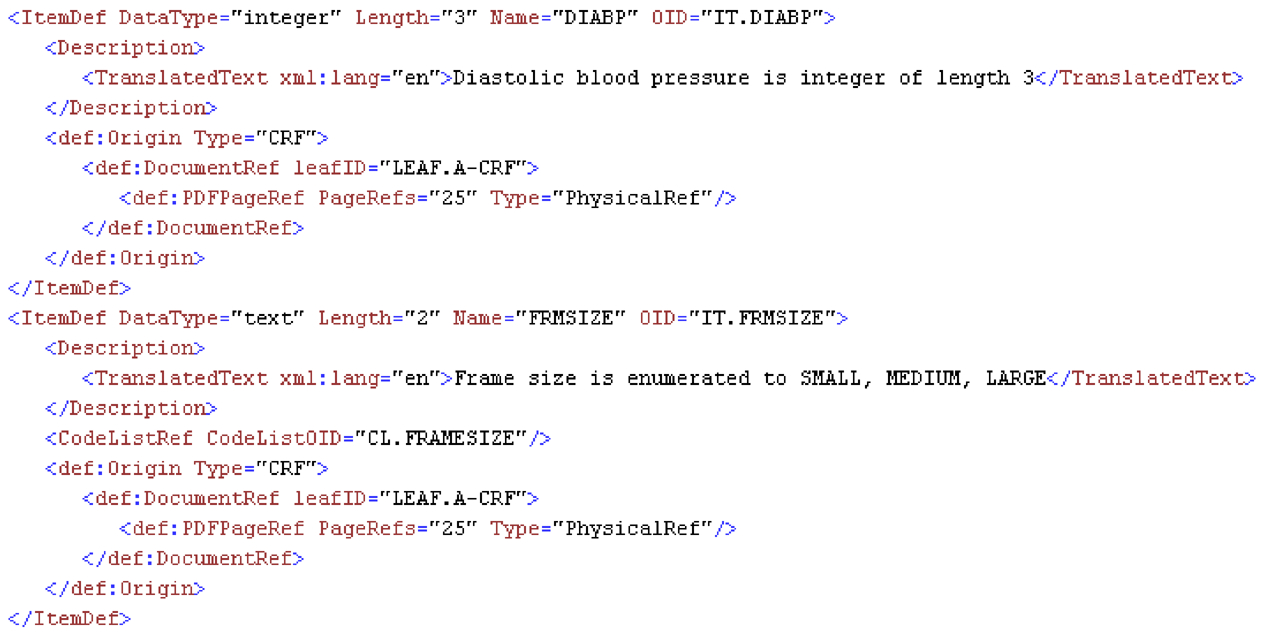
```
<ItemDef DataType="text" Length="80" Name="VSORRES" OID="VS.VSORRES">
   <Description>
      <TranslatedText xml:lang="en">Result or Finding in Original Units</TranslatedText>
   </Description>
   <def:ValueListRef ValueListOID="VL.VSORRES"/>
</ItemDef>
```

and:



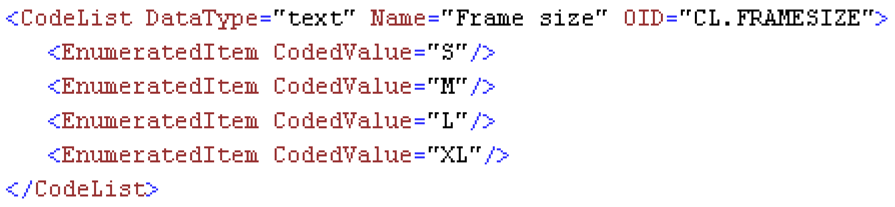
```
<def:ValueListDef OID="VL.VSORRES">
   <ItemRef ItemOID="IT.DIABP" Mandatory="No" OrderNumber="1">
      <def:WhereClauseRef WhereClauseOID="WC.IT.DIABP"/>
   </ItemRef>
   <ItemRef ItemOID="IT.FRMSIZE" Mandatory="No" OrderNumber="2">
      <def:WhereClauseRef WhereClauseOID="WC.IT.FRMSIZE"/>
   </ItemRef>
   <ItemRef ItemOID="IT.HEIGHT" Mandatory="No" OrderNumber="3">
      <def:WhereClauseRef WhereClauseOID="WC.IT.HEIGHT"/>
   </ItemRef>
   <ItemRef ItemOID="IT.SYSBP" Mandatory="No" OrderNumber="4">
      <def:WhereClauseRef WhereClauseOID="WC.IT.SYSBP"/>
   </ItemRef>
   <ItemRef ItemOID="IT.WEIGHT" Mandatory="No" OrderNumber="5">
      <def:WhereClauseRef WhereClauseOID="WC.IT.WEIGHT"/>
   </ItemRef>
</def:ValueListDef>
<def:WhereClauseDef OID="WC.IT.DIABP">
   <RangeCheck Comparator="EQ" SoftHard="Soft" def:ItemOID="VS.VSTESTCD">
      <CheckValue>DIABP</CheckValue>
   </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.IT.FRMSIZE">
   <RangeCheck Comparator="EQ" SoftHard="Soft" def:ItemOID="VS.VSTESTCD">
      <CheckValue>FRMSIZE</CheckValue>
   </RangeCheck>
```

The first picture showing how the ValueList VL.VSORRES is referenced by VSORRES, the second one showing the details and some (the first two from five) "WhereClause" definitions. The details are then found in the corresponding "ItemDefs":



```
<ItemDef DataType="integer" Length="3" Name="DIABP" OID="IT.DIABP">
   <Description>
      <TranslatedText xml:lang="en">Diastolic blood pressure is integer of length 3</TranslatedText>
   </Description>
   <def:Origin Type="CRF">
      <def:DocumentRef leafID="LEAF.A-CRF">
         <def:PDFPageRef PageRefs="25" Type="PhysicalRef"/>
      </def:DocumentRef>
   </def:Origin>
</ItemDef>
<ItemDef DataType="text" Length="2" Name="FRMSIZE" OID="IT.FRMSIZE">
   <Description>
      <TranslatedText xml:lang="en">Frame size is enumerated to SMALL, MEDIUM, LARGE</TranslatedText>
   </Description>
   <CodeListRef CodeListOID="CL.FRAMESIZE"/>
   <def:Origin Type="CRF">
      <def:DocumentRef leafID="LEAF.A-CRF">
         <def:PDFPageRef PageRefs="25" Type="PhysicalRef"/>
      </def:DocumentRef>
   </def:Origin>
</ItemDef>
```

One nicely sees that it is stated that for DIABP, an integer is expected, whereas for FRMSIZE, a reference is made to the codelist CL.SIZE, listing the enumerations "S" (small), "M" (medium), and "L" (large) and "XL" (extra large):

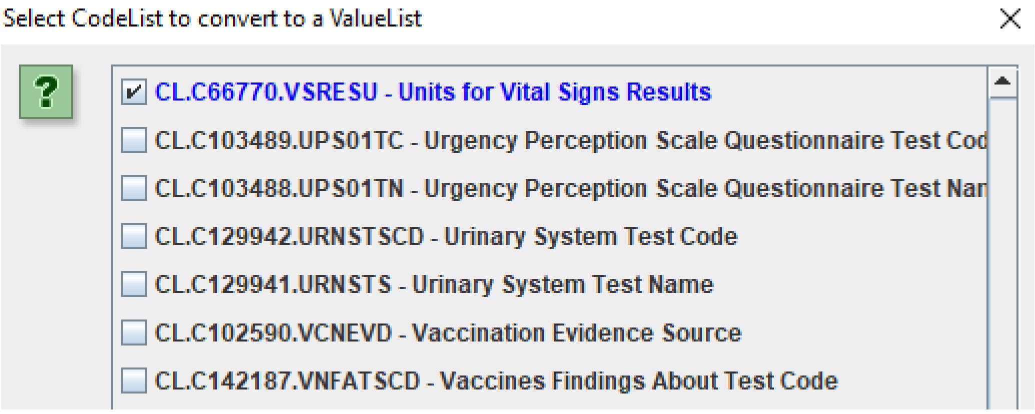


```
<CodeList DataType="text" Name="Frame size" OID="CL.FRAMESIZE">
   <EnumeratedItem CodedValue="S"/>
   <EnumeratedItem CodedValue="M"/>
   <EnumeratedItem CodedValue="L"/>
   <EnumeratedItem CodedValue="XL"/>
</CodeList>
```

Creating a ValueList for VSORRESU

Where creating a valuelist for VSORRES (vital signs original result) was pretty simple, we will make us ourselves a lot more difficult for VSORRESU (vital signs original results units). We will define that in case the weight of the subject was captured at a site in the USA, then the unit in VSORRESU is "pounds" whereas when the weight was captured at a site in Germany or France, the unit used is "kilograms". We will also state that the unit used is "mmHg" when the test code is either DIABP or SYSBP.

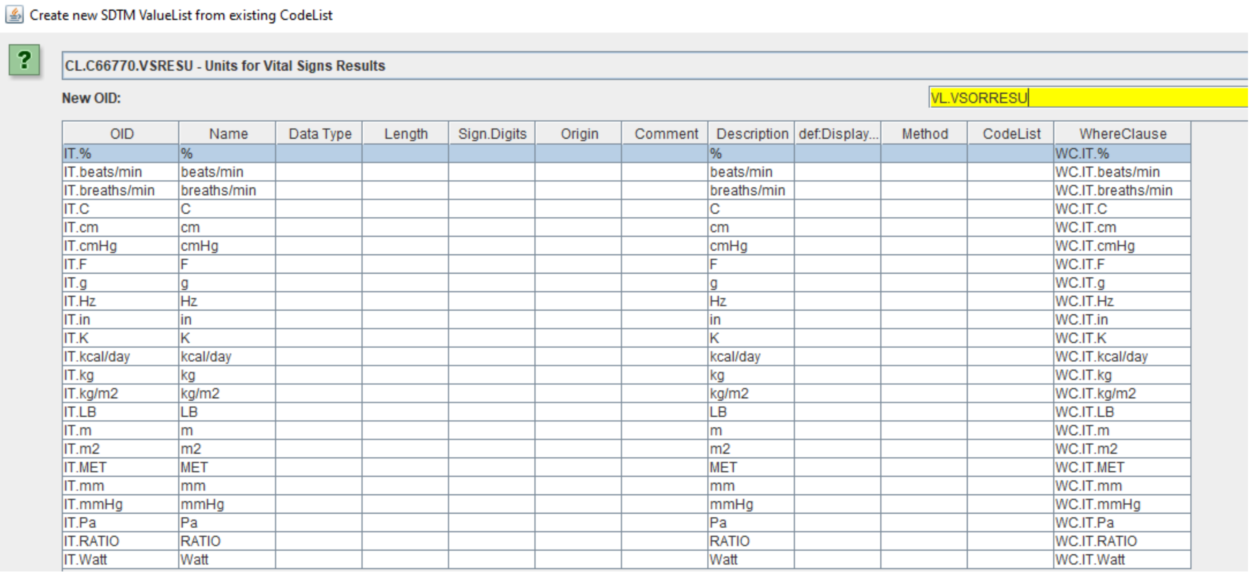
Also here, we can start from an existing codelist and then transform that to a valuelist using the menu "Insert – Create new ValueList from existing CodeList":



Select CodeList to convert to a ValueList

☑ CL.C66770.VSRESU - Units for Vital Signs Results
□ CL.C103489.UPS01TC - Urgency Perception Scale Questionnaire Test Cod
□ CL.C103488.UPS01TN - Urgency Perception Scale Questionnaire Test Nar
□ CL.C129942.URNSTSCD - Urinary System Test Code
□ CL.C129941.URNSTS - Urinary System Test Name
□ CL.C102590.VCNEVD - Vaccination Evidence Source
□ CL.C142187.VNFATSCD - Vaccines Findings About Test Code

Remark that the choice of the codelist may depend on which version of controlled terminology you use: CDISC has not been very consequent here.
This time we select the "Unit for Vital Signs Results" codelist, which is then transformed to a ValueList:



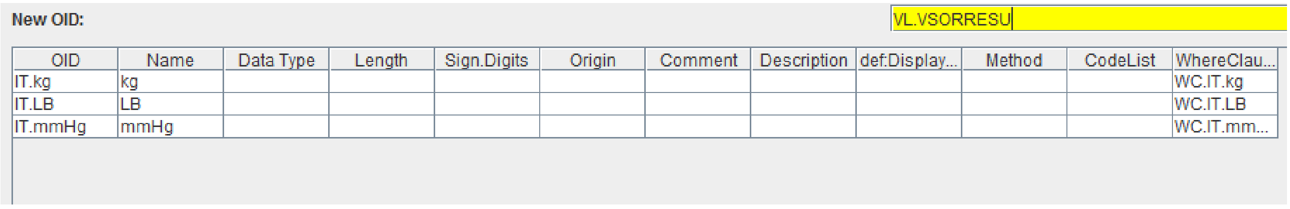
Create new SDTM ValueList from existing CodeList

CL.C66770.VSRESU - Units for Vital Signs Results

New OID: VL.VSORRESU

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:Display...	Method	CodeList	WhereClause
IT.%	%						%				WC.IT.%
IT.beats/min	beats/min						beats/min				WC.IT.beats/min
IT.breaths/min	breaths/min						breaths/min				WC.IT.breaths/min
IT.C	C						C				WC.IT.C
IT.cm	cm						cm				WC.IT.cm
IT.cmHg	cmHg						cmHg				WC.IT.cmHg
IT.F	F						F				WC.IT.F
IT.g	g						g				WC.IT.g
IT.Hz	Hz						Hz				WC.IT.Hz
IT.in	in						in				WC.IT.in
IT.K	K						K				WC.IT.K
IT.kcal/day	kcal/day						kcal/day				WC.IT.kcal/day
IT.kg	kg						kg				WC.IT.kg
IT.kg/m2	kg/m2						kg/m2				WC.IT.kg/m2
IT.LB	LB						LB				WC.IT.LB
IT.m	m						m				WC.IT.m
IT.m2	m2						m2				WC.IT.m2
IT.MET	MET						MET				WC.IT.MET
IT.mm	mm						mm				WC.IT.mm
IT.mmHg	mmHg						mmHg				WC.IT.mmHg
IT.Pa	Pa						Pa				WC.IT.Pa
IT.RATIO	RATIO						RATIO				WC.IT.RATIO
IT.Watt	Watt						Watt				WC.IT.Watt

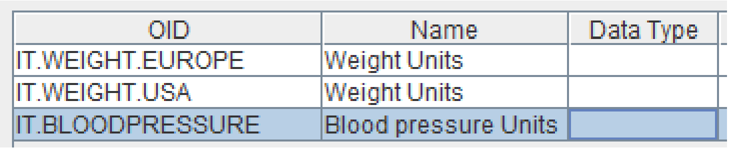
Renaming it to "VL.VSORRESU" and removing all unnecessary rows:



New OID: VL.VSORRESU

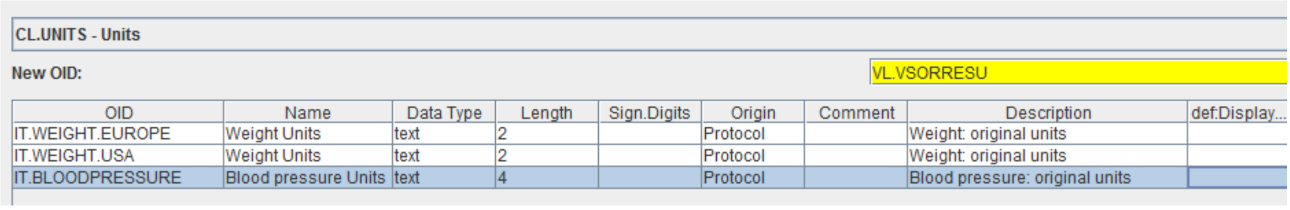
OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:Display...	Method	CodeList	WhereClau...
IT.kg	kg										WC.IT.kg
IT.LB	LB										WC.IT.LB
IT.mmHg	mmHg										WC.IT.mm...

Of course we can also alter the OID and Name of the items, e.g.:



OID	Name	Data Type
IT.WEIGHT.EUROPE	Weight Units	
IT.WEIGHT.USA	Weight Units	
IT.BLOODPRESSURE	Blood pressure Units	

as the unit itself is given, the data type is "text" in all cases, and again, we can state that the origin is a specific page of the CRF. In this case, it could however also "protocol defined", as the source might simply have been the protocol stating "weight must be measured in pounds in the US, and in kilograms in France and Germany". We will assume the latter:

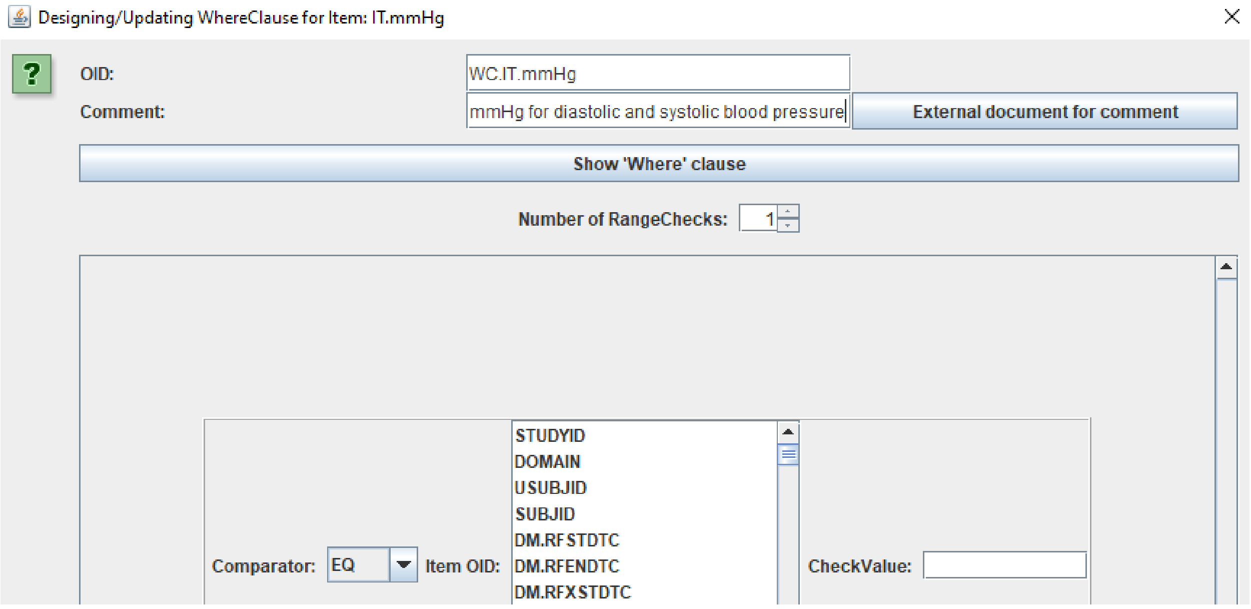


CL.UNITS - Units

New OID: VL.VSORRESU

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:Display...
IT.WEIGHT.EUROPE	Weight Units	text	2		Protocol		Weight: original units	
IT.WEIGHT.USA	Weight Units	text	2		Protocol		Weight: original units	
IT.BLOODPRESSURE	Blood pressure Units	text	4		Protocol		Blood pressure: original units	

also having assigned a short description. Now we still need to define under which conditions each of these should be used ("Where Clause"). For "blood pressure" we need to state that mmHg needs to be used when VSTESTCD=DIABP or VSTESTCD=DIABP. This is pretty easy to define:



Designing/Updating WhereClause for Item: IT.mmHg

OID: WC.IT.mmHg

Comment: mmHg for diastolic and systolic blood pressure

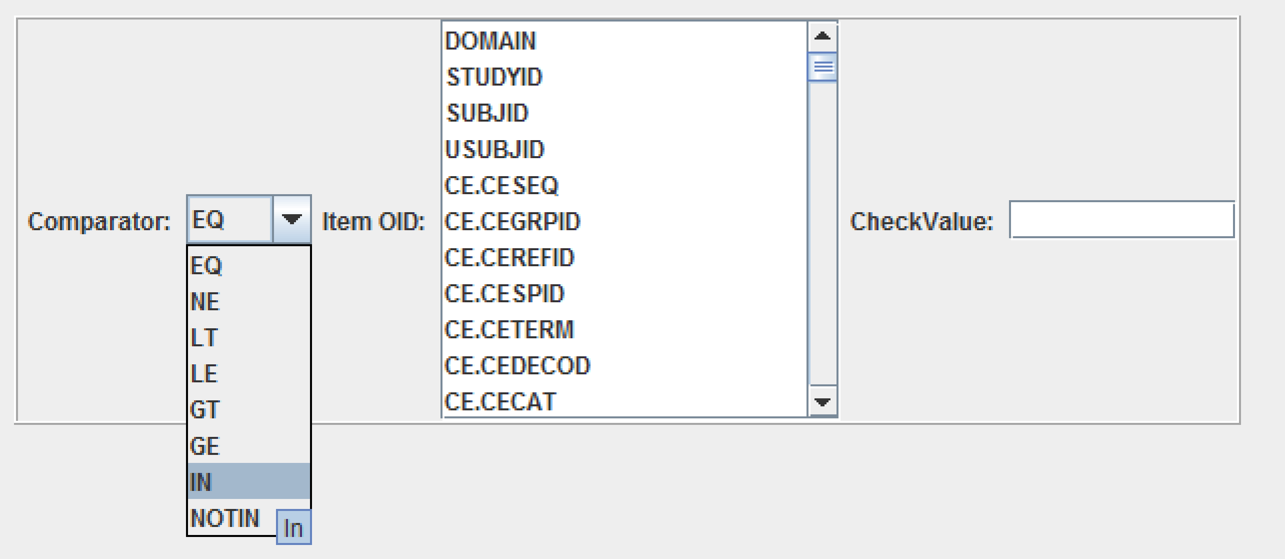
Number of RangeChecks: 1

Comparator: EQ

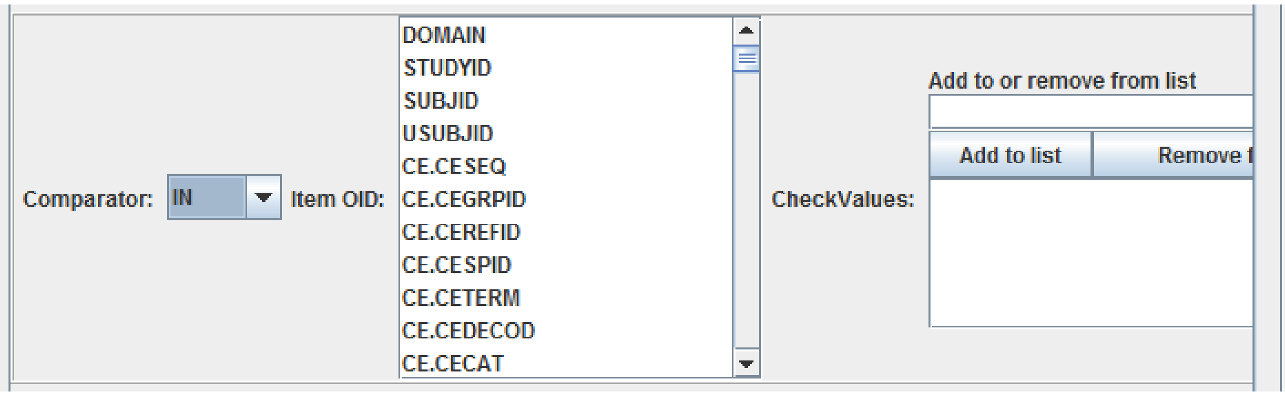
Item OID: STUDYID, DOMAIN, USUBJID, SUBJID, DM.RFSTDTC, DM.RFENDTC, DM.RFXSTDTC

CheckValue:

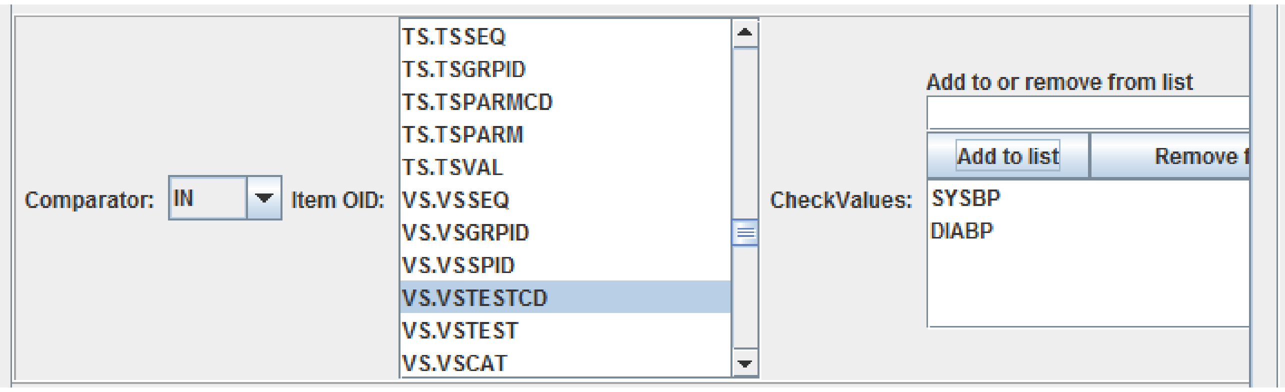
We cannot simply state VSTESTCD=DIABP as we also need to state "or VSTESTCD=SYSBP". The simplest way we can do this is to select the comparator "IN", meaning "in the set of ..":



leading to a somewhat different screen:



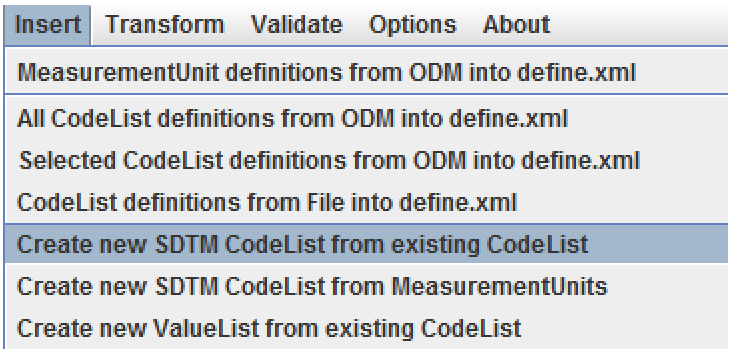
On the right, we can now use items, like "DIABP" and "SYSBP" and selecting VS.VSTESTCD in the middle ("ItemOID"). Using the "Add to list" button, this leads to:



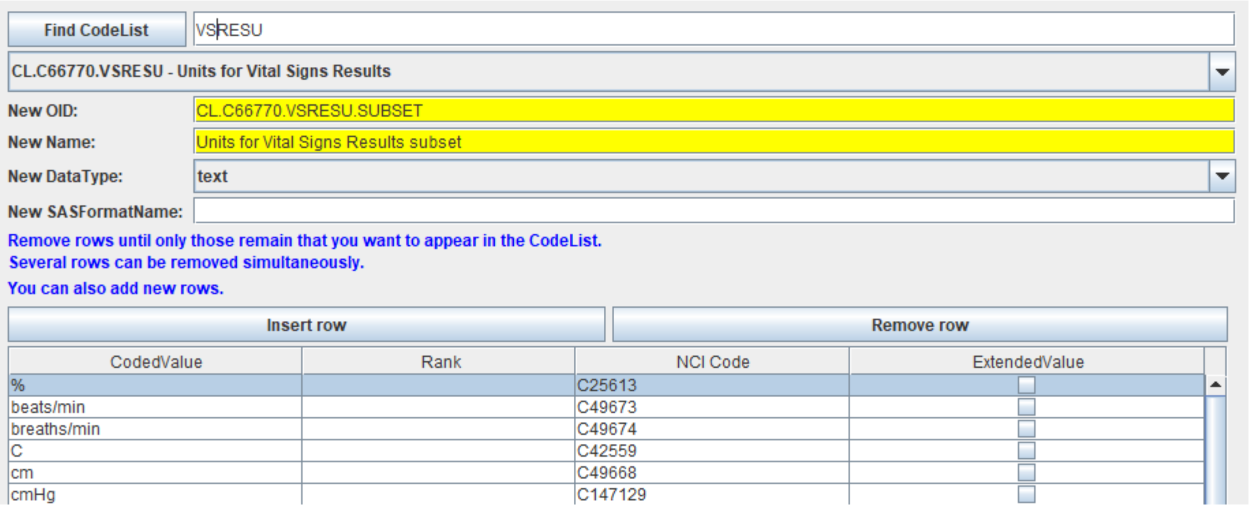
where we have stated that "the condition for using mmHg is that the value of VSTESTCD is in the set {"SYSBP",DIABP"}", which is exactly what we want.

Now there is one important thing that we forgot: "what is the value that is expected" in VSORRESU itself? The only allowed one is "mmHg", so we need to generate a CodeList with only one value which is "mmHg".

We will leave the ValueList definitions for a moment, click OK until we are back in the main screen, and generate a new CodeList only containing "mmHg". To do this use the menu "Insert – Create new CodeList from existing CodeList":


Insert
Transform
Validate
Options
About
MeasurementUnit definitions from ODM into define.xml
All CodeList definitions from ODM into define.xml
Selected CodeList definitions from ODM into define.xml
CodeList definitions from File into define.xml
Create new SDTM CodeList from existing CodeList
Create new SDTM CodeList from MeasurementUnits
Create new ValueList from existing CodeList

and select CL.VSRESU for which we will make a sub-codelist:



VSRESU

CL.C66770.VSRESU - Units for Vital Signs Results

New OID: CL.C66770.VSRESU.SUBSET

New Name: Units for Vital Signs Results subset

New DataType: text

New SASFormatName:

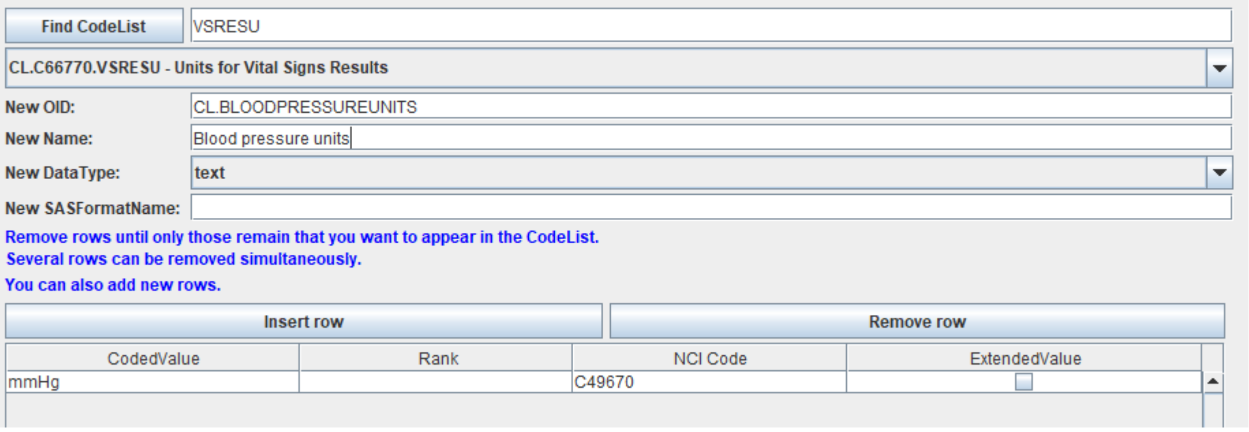
Remove rows until only those remain that you want to appear in the CodeList.
Several rows can be removed simultaneously.
You can also add new rows.

CodedValue	Rank	NCI Code	ExtendedValue
%		C25613	□
beats/min		C49673	□
breaths/min		C49674	□
C		C42559	□
cm		C49668	□
cmHg		C147129	□

which then displays the full list of units available for vital signs.

We now need to remove all unecessary rows until we only keep "mmHg"[5], and give the new codelist the OID "CL.BLOODPRESSUREUNITS", as it keeps all the units (well, only one in this case) for measurements of blood pressure:

[5] The corresponding international UCUM unit is however "mm[Hg]"



VSRESU

CL.C66770.VSRESU - Units for Vital Signs Results

New OID: CL.BLOODPRESSUREUNITS

New Name: Blood pressure units

New DataType: text

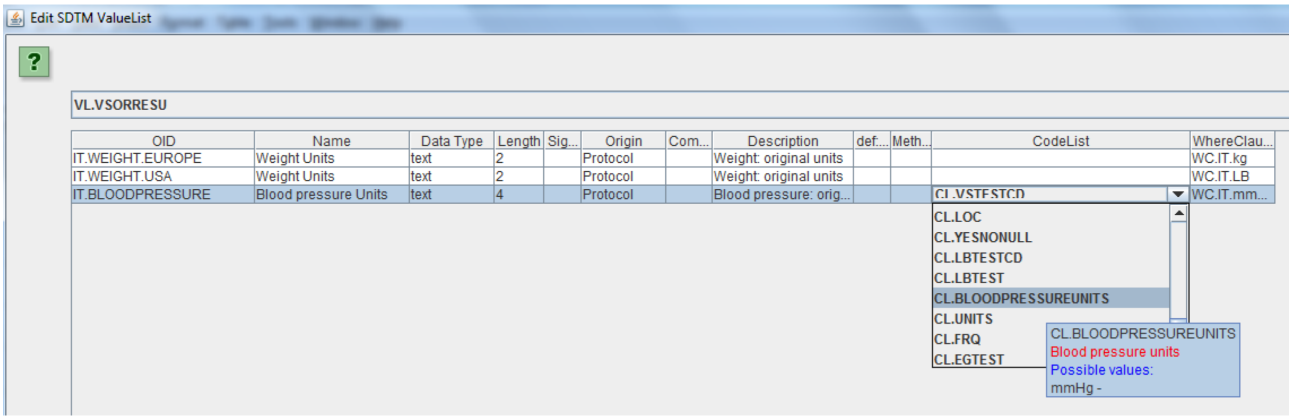
New SASFormatName:

Remove rows until only those remain that you want to appear in the CodeList.
Several rows can be removed simultaneously.
You can also add new rows.

CodedValue	Rank	NCI Code	ExtendedValue
mmHg		C49670	□

There will be other cases where the sub-codelist can contain more than one entry. Suppose for example that the US-based investigators always report "HEIGHT" in "inches", whereas some of their colleagues report in centimeters and others in meters. In Europe, these investigators have tickboxes for "m" and "cm" on the CRF, from which they can choose.
As such, there is no way that we can define a "WhereClause" for distinguishing between "cm" and "m" as it depends on the investigators habit (and may of the mood of the day...).

We can now go back to the ValueList VL.VSORRESU that we were editing and now assign the CodeList CL.BLOODPRESSUREUNITS to the item "Blood pressure units" (IT.BLOODPRESSURE):



VL.VSORRESU

OID	Name	Data Type	Length	Sig...	Origin	Com...	Description	def:...	Meth...	CodeList	WhereClau...
IT.WEIGHT.EUROPE	Weight Units	text	2		Protocol		Weight: original units				WC.IT.kg
IT.WEIGHT.USA	Weight Units	text	2		Protocol		Weight: original units				WC.IT.LB
IT.BLOODPRESSURE	Blood pressure Units	text	4		Protocol		Blood pressure: orig...			CL.VSTESTCD	WC.IT.mm...

CL.LOC
CL.YESNONULL
CL.LBTESTCD
CL.LBTEST
CL.BLOODPRESSUREUNITS
CL.UNITS
CL.FRQ
CL.EGTEST

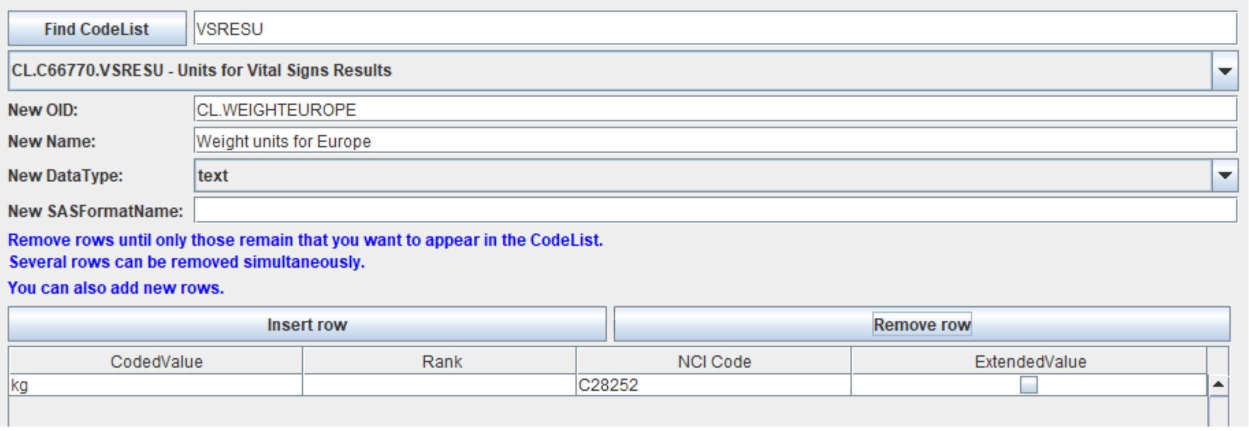
CL.BLOODPRESSUREUNITS
Blood pressure units
Possible values:
mmHg -

The "Where Clause" was already added stating that the condition is VSTESTC in "IN" the set of {"DIABP", "SYSBP"}.

We now still need to repeat this for the use of weight units in the USA (IT.WEIGHTUSA) and in Europe (IT.WEIGHTEUROPE"). The exact values for the OIDs do not care[6], as long as we choose the correct attributes such as data type, origin, codelist, and then add a well-defined "WhereClause".

For the item "IT.WEIGHTEUROPE" we once again need to create a codelist which only contains a single entry "kg" (kilograms), e.g.:

[6] But choosing something meaningful may help in doing selections later, especially when there are many of them. OIDs are fully arbitrary ...



VSRESU

CL.C66770.VSRESU - Units for Vital Signs Results

New OID: CL.WEIGHTEUROPE

New Name: Weight units for Europe

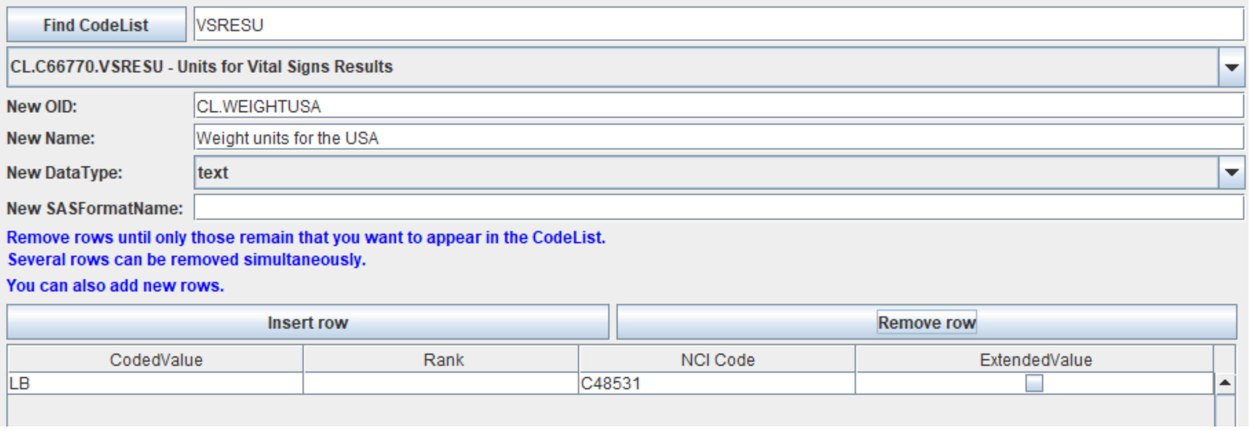
New DataType: text

New SASFormatName:

Remove rows until only those remain that you want to appear in the CodeList.
Several rows can be removed simultaneously.
You can also add new rows.

CodedValue	Rank	NCI Code	ExtendedValue
kg		C28252	□

and a similar one but with different contents for "Weight units in the USA":



VSRESU

CL.C66770.VSRESU - Units for Vital Signs Results

New OID: CL.WEIGHTUSA

New Name: Weight units for the USA

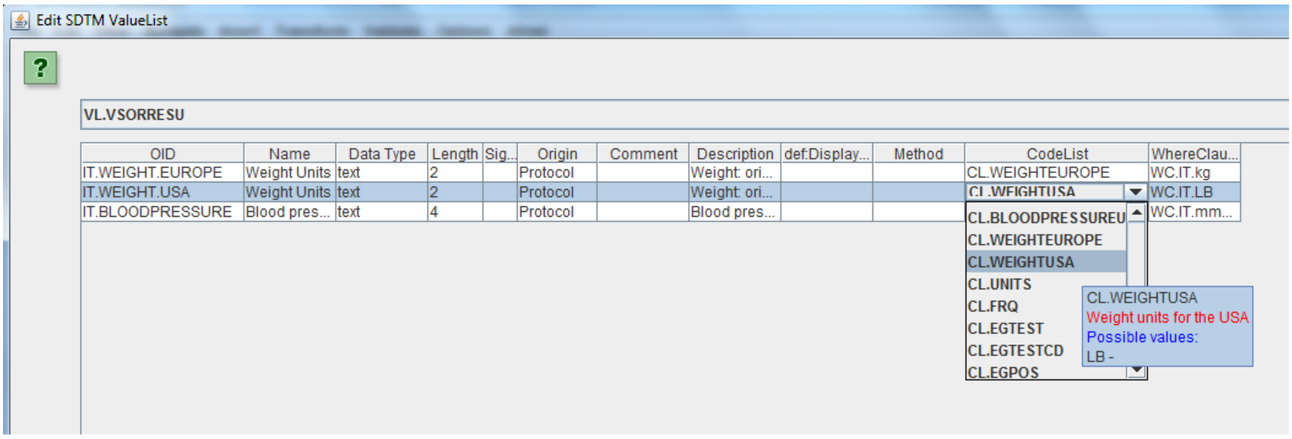
New DataType: text

New SASFormatName:

Remove rows until only those remain that you want to appear in the CodeList.
Several rows can be removed simultaneously.
You can also add new rows.

CodedValue	Rank	NCI Code	ExtendedValue
LB		C48531	□

and then assign them to their corresponding items in the valuelist:

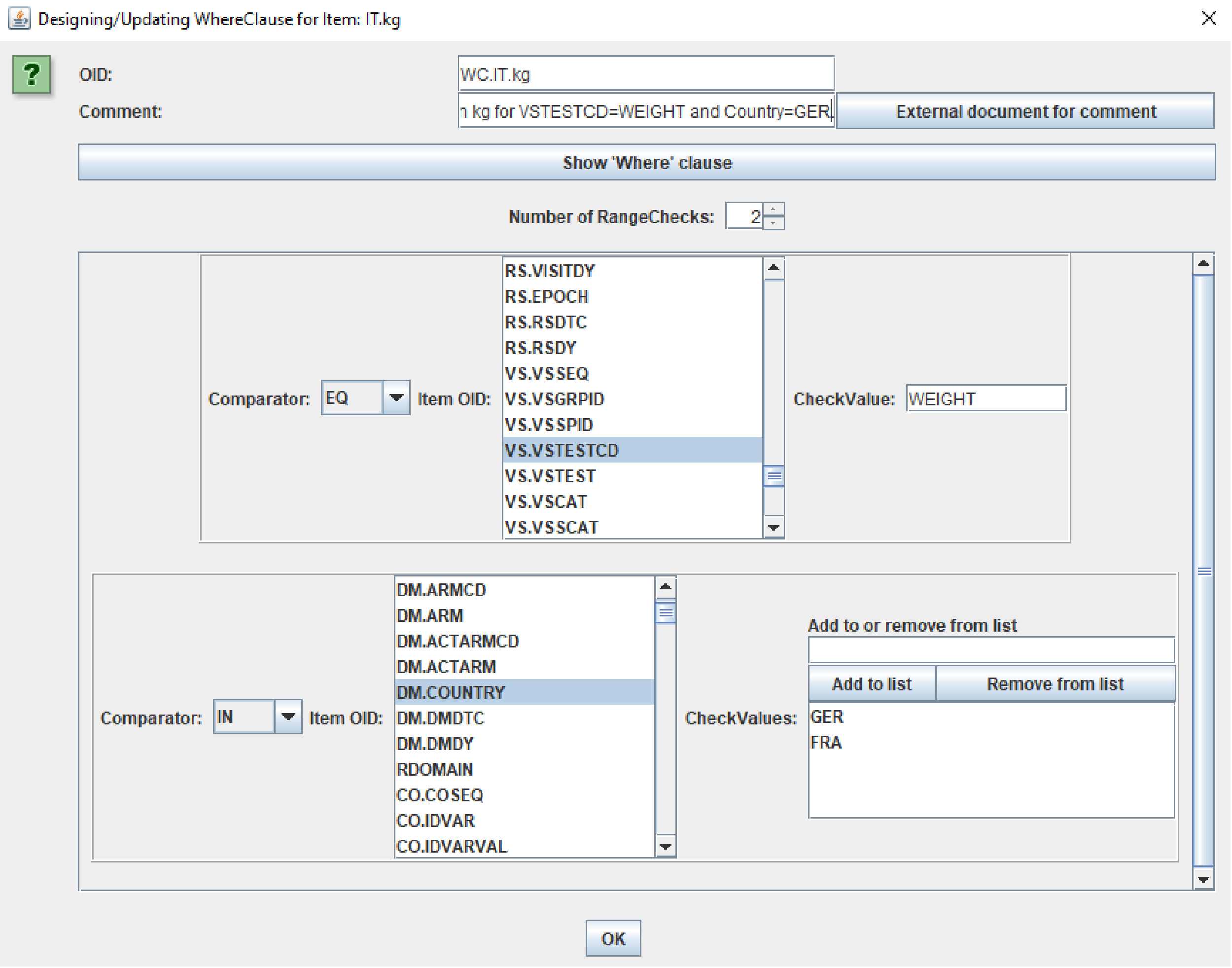


VL.VSORRESU

OID	Name	Data Type	Length	Sig...	Origin	Comment	Description	def:Display...	Method	CodeList	WhereClau...
IT.WEIGHT.EUROPE	Weight Units	text	2		Protocol		Weight: ori...			CL.WEIGHTEUROPE	WC.IT.kg
IT.WEIGHT.USA	Weight Units	text	2		Protocol		Weight: ori...			CL.WEIGHTUSA	WC.IT.LB
IT.BLOODPRESSURE	Blood pres...	text	4		Protocol		Blood pres...				WC.IT.mm...

CL.WEIGHTUSA
Weight units for the USA
Possible values:
LB -

All we now still need to do is to assign a "WhereClause" for both. Let's start with "Weight units in Europe". The first condition is of course that VSTESTCD=WEIGHT, the second being that the country is either Germany or France. The "WhereClause" editor than looks like:



Designing/Updating WhereClause for Item: IT.kg

OID: WC.IT.kg

Comment: n kg for VSTESTCD=WEIGHT and Country=GER

Number of RangeChecks: 2

Comparator: EQ Item OID: VS.VSTESTCD CheckValue: WEIGHT

RS.VISITDY
RS.EPOCH
RS.RSDTC
RS.RSDY
VS.VSSEQ
VS.VSGRPID
VS.VSSPID
VS.VSTESTCD
VS.VSTEST
VS.VSCAT
VS.VSSCAT

Comparator: IN Item OID: DM.COUNTRY CheckValues: GER, FRA

DM.ARMCD
DM.ARM
DM.ACTARMCD
DM.ACTARM
DM.COUNTRY
DM.DMDTC
DM.DMDY
RDOMAIN
CO.COSEQ
CO.IDVAR
CO.IDVARVAL

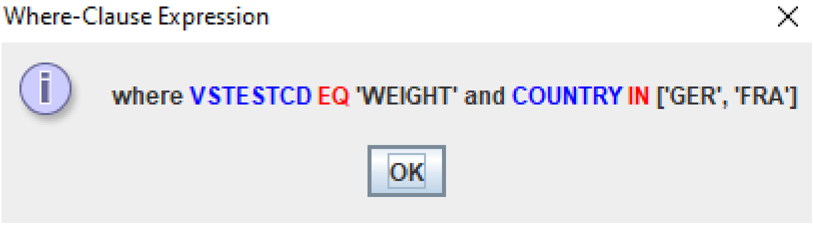
having two checks:

VSTESTCD=WEIGHT

and:

DM.COUNTRY has either the value "GER" (Germany) or "FRA" (France).

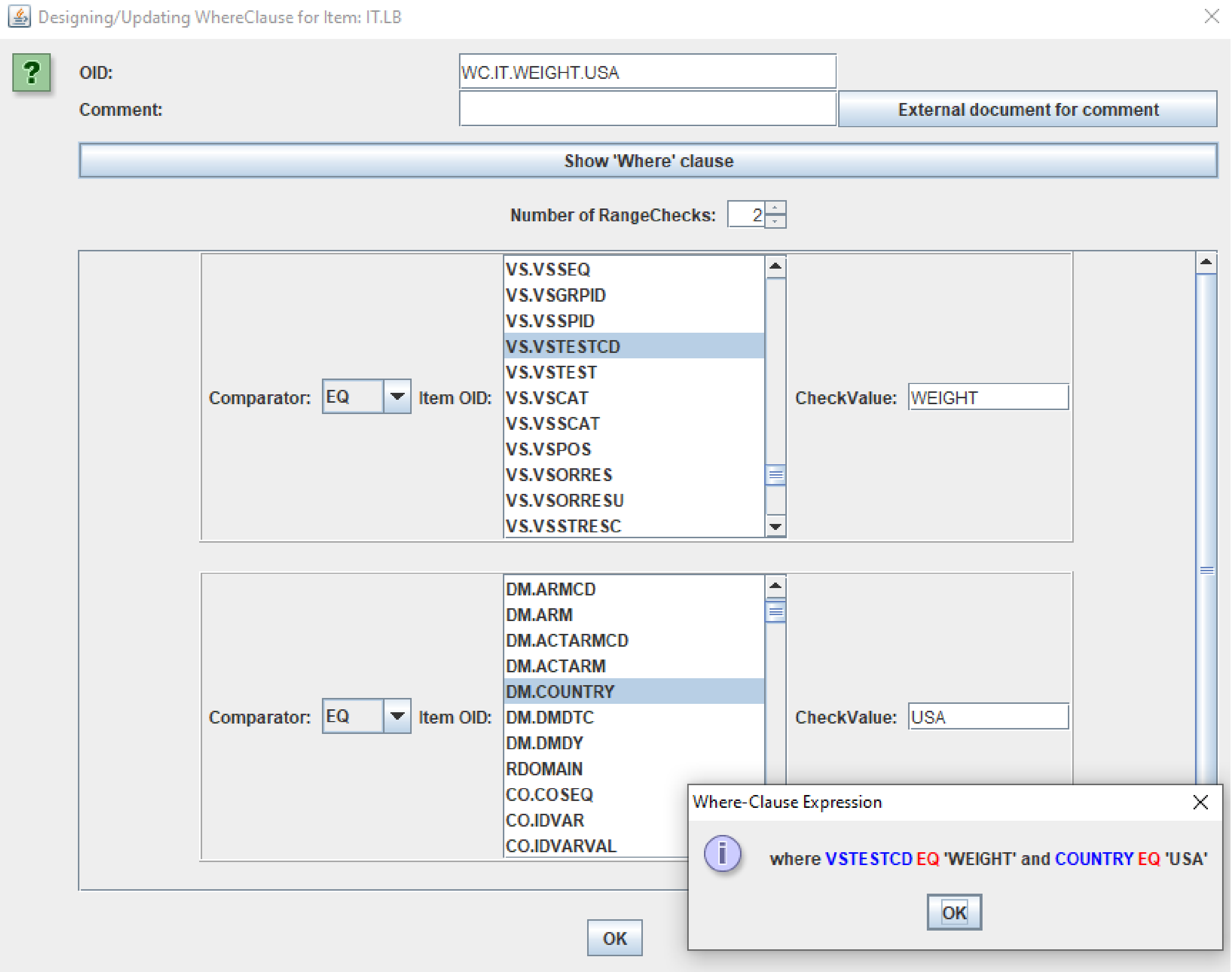
One can again use the button "Show 'Where' clause":



Where-Clause Expression

where VSTESTCD EQ 'WEIGHT' and COUNTRY IN ['GER', 'FRA']

The "WhereClause" for weight units in the USA is somewhat simpler:



Designing/Updating WhereClause for Item: IT.LB

OID: WC.IT.WEIGHT.USA

Comment:

Number of RangeChecks: 2

Comparator: EQ Item OID: VS.VSTESTCD CheckValue: WEIGHT

Comparator: EQ Item OID: DM.COUNTRY CheckValue: USA

Where-Clause Expression

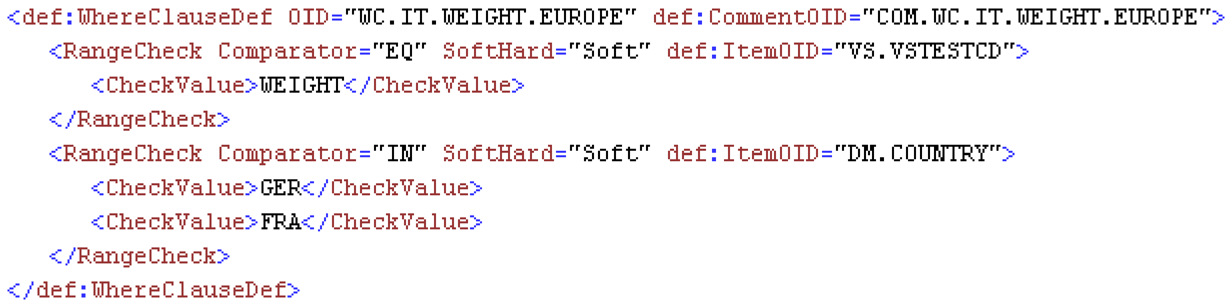
where VSTESTCD EQ 'WEIGHT' and COUNTRY EQ 'USA'

defining two conditions:
VSTESTCD=WEIGHT
and:
DM.COUNTRY=USA

In the underlying define.xml, this is reflected by:

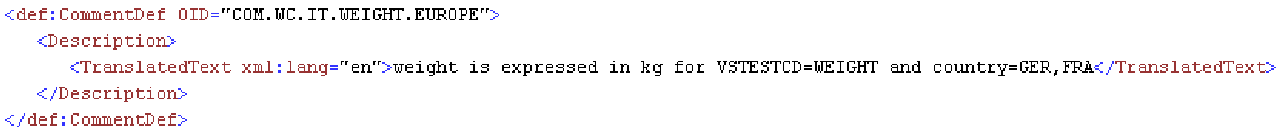
```
<def:ValueListDef OID="VL.VSORRESU">
   <ItemRef ItemOID="IT.WEIGHT.EUROPE" Mandatory="No" OrderNumber="1">
      <def:WhereClauseRef WhereClauseOID="WC.IT.WEIGHT.EUROPE"/>
   </ItemRef>
   <ItemRef ItemOID="IT.WEIGHT.USA" Mandatory="No" OrderNumber="2">
      <def:WhereClauseRef WhereClauseOID="WC.IT.WEIGHT.USA"/>
   </ItemRef>
   <ItemRef ItemOID="IT.BLOODPRESSURE" Mandatory="No" OrderNumber="3">
      <def:WhereClauseRef WhereClauseOID="WC.IT.mmHg"/>
   </ItemRef>
</def:ValueListDef>
<def:WhereClauseDef OID="WC.IT.BMI" def:CommentOID="COM.WC.IT.BMI">
   <RangeCheck Comparator="EQ" SoftHard="Soft" def:ItemOID="VS.VSTESTCD">
      <CheckValue>BMI</CheckValue>
   </RangeCheck>
</def:WhereClauseDef>
```

and e.g. the "WhereClause" for units for weight in Germany and France:



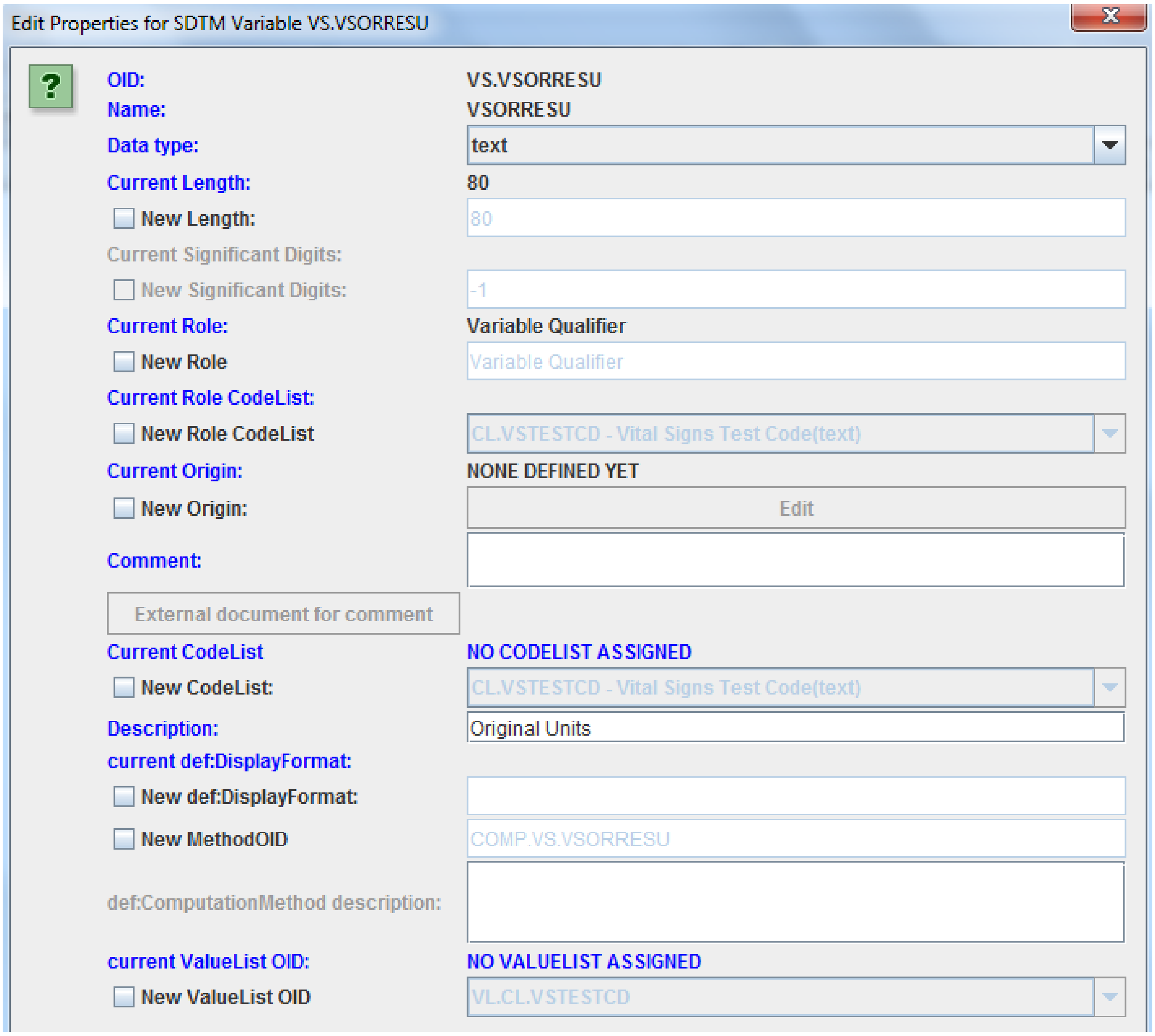
```
<def:WhereClauseDef OID="WC.IT.WEIGHT.EUROPE" def:CommentOID="COM.WC.IT.WEIGHT.EUROPE">
   <RangeCheck Comparator="EQ" SoftHard="Soft" def:ItemOID="VS.VSTESTCD">
      <CheckValue>WEIGHT</CheckValue>
   </RangeCheck>
   <RangeCheck Comparator="IN" SoftHard="Soft" def:ItemOID="DM.COUNTRY">
      <CheckValue>GER</CheckValue>
      <CheckValue>FRA</CheckValue>
   </RangeCheck>
</def:WhereClauseDef>
```

and the corresponding comment:



```
<def:CommentDef OID="COM.WC.IT.WEIGHT.EUROPE">
   <Description>
      <TranslatedText xml:lang="en">weight is expressed in kg for VSTESTCD=WEIGHT and country=GER,FRA</TranslatedText>
   </Description>
</def:CommentDef>
```

Not a bad idea to check whether we really assigned our newly developed valuelist to VSORRESU:



Edit Properties for SDTM Variable VS.VSORRESU

OID: VS.VSORRESU

Name: VSORRESU

Data type: text

Current Length: 80

☐ New Length: 80

Current Significant Digits:

☐ New Significant Digits: -1

Current Role: Variable Qualifier

☐ New Role Variable Qualifier

Current Role CodeList:

☐ New Role CodeList CL.VSTESTCD - Vital Signs Test Code(text)

Current Origin: NONE DEFINED YET

☐ New Origin: Edit

Comment:

External document for comment

Current CodeList NO CODELIST ASSIGNED

☐ New CodeList: CL.VSTESTCD - Vital Signs Test Code(text)

Description: Original Units

current def:DisplayFormat:

☐ New def:DisplayFormat:

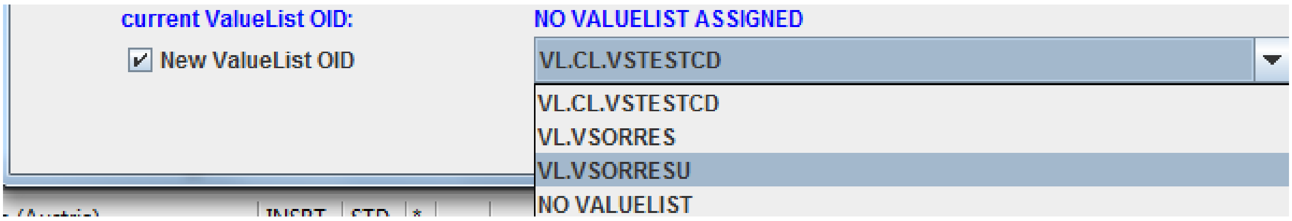
☐ New MethodOID COMP.VS.VSORRESU

def:ComputationMethod description:

current ValueList OID: NO VALUELIST ASSIGNED

☐ New ValueList OID VL.CL.VSTESTCD

It looks as we didn't, so we do it now using the checkbox "New ValueList OID":

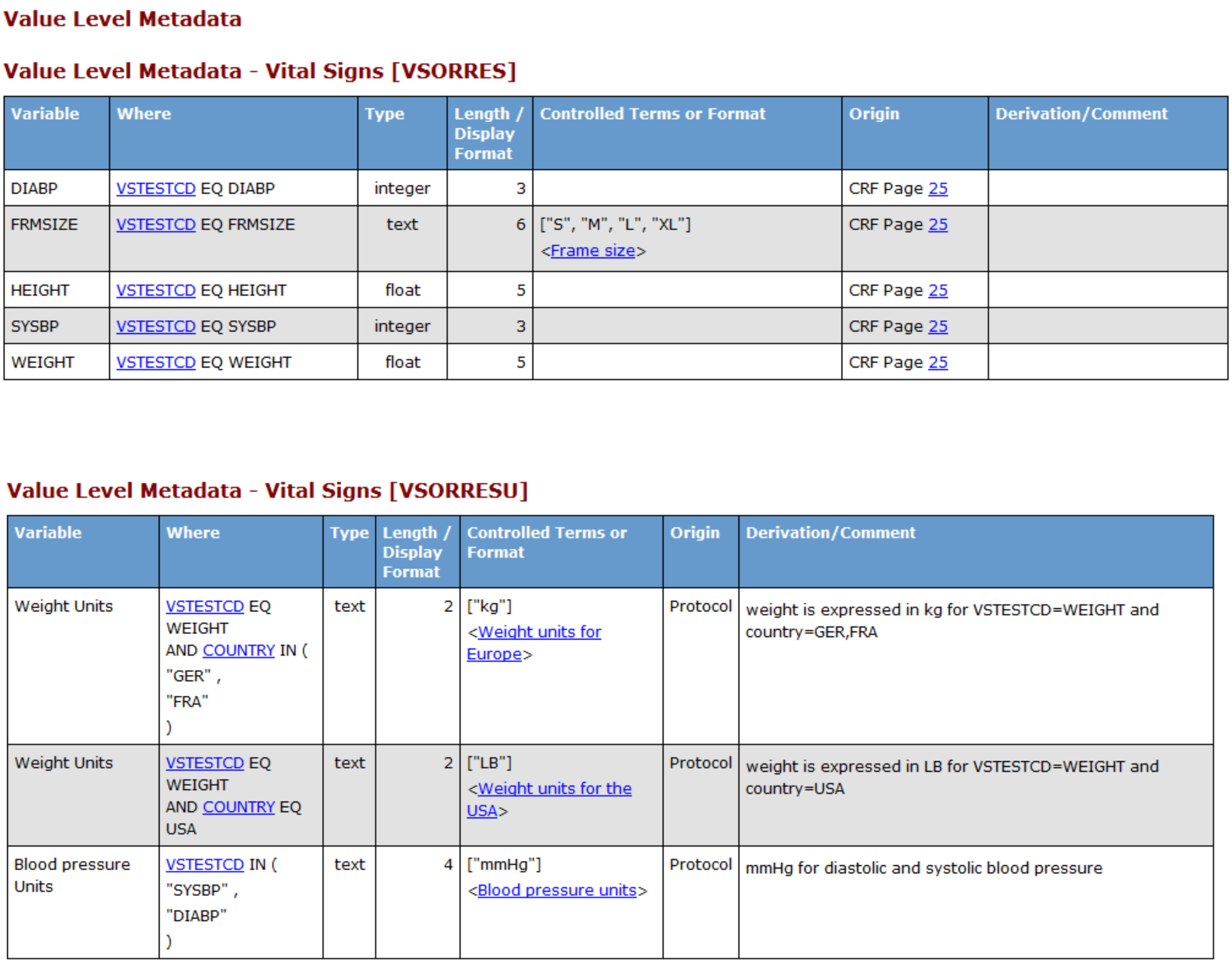

current ValueList OID: NO VALUELIST ASSIGNED

☑ New ValueList OID: VL.CL.VSTESTCD

which appears in the define.xml as:

```
<ItemDef DataType="text" Length="6" Name="VSORRESU" OID="VS.VSORRESU">
   <Description>
      <TranslatedText xml:lang="en">Original Units</TranslatedText>
   </Description>
   <def:ValueListRef ValueListOID="VL.VSORRESU"/>
</ItemDef>
```

and in the HTML view (use "View – View define.ml in browser):


Value Level Metadata

Value Level Metadata - Vital Signs [VSORRES]

Variable	Where	Type	Length / Display Format	Controlled Terms or Format	Origin	Derivation/Comment
DIABP	VSTESTCD EQ DIABP	integer	3		CRF Page 25	
FRMSIZE	VSTESTCD EQ FRMSIZE	text	6	["S", "M", "L", "XL"] <Frame size>	CRF Page 25	
HEIGHT	VSTESTCD EQ HEIGHT	float	5		CRF Page 25	
SYSBP	VSTESTCD EQ SYSBP	integer	3		CRF Page 25	
WEIGHT	VSTESTCD EQ WEIGHT	float	5		CRF Page 25	

Value Level Metadata - Vital Signs [VSORRESU]

Variable	Where	Type	Length / Display Format	Controlled Terms or Format	Origin	Derivation/Comment
Weight Units	VSTESTCD EQ WEIGHT AND COUNTRY IN ("GER" , "FRA")	text	2	["kg"] <Weight units for Europe>	Protocol	weight is expressed in kg for VSTESTCD=WEIGHT and country=GER,FRA
Weight Units	VSTESTCD EQ WEIGHT AND COUNTRY EQ USA	text	2	["LB"] <Weight units for the USA>	Protocol	weight is expressed in LB for VSTESTCD=WEIGHT and country=USA
Blood pressure Units	VSTESTCD IN ("SYSBP" , "DIABP")	text	4	["mmHg"] <Blood pressure units>	Protocol	mmHg for diastolic and systolic blood pressure

This finalizes our tutorial about the use of "ValueList" and the "Where Clause". Once again, when using "Where Clause" in define.xml 2.0/2.1, you are not allowed to use nested value levels anymore.

We also showed how easy it is using SDTM-ETL to define the "Where Clauses" in a user-friendly way using a graphical, easy-to-use dialog, and without any necessity to do any XML editing or to "fight" with "black box" worksheets and software.