SDTM-ETL 3.1 User Manual and Tutorial

Author: Jozef Aerts, XML4Pharma

Last update: 2014-08-23



Working with the WhereClause in define.xml 2.0

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 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. 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 <u>define.xml 2.0 specification</u>: 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 Germany¹. 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 nestes "ValueLists" instead. In define.xml 2.0, 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 is "pounds" when VSTESTCD=WEIGHT and DM.COUNTRY=USA
- the unit 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. We can either start from an existing codelist (Insert – Create New ValueList from existing CodeList", leading to:

Selected C	odeLists to convert to ValueLists	X
?	 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 TESTCD. This leads to a new dialog with:

Name BMI BODYFAT BSA DIABP	Data Type	Length	Qian Dialla				VL.CL.VSTE	STCD		
Name BMI BODYFAT BSA DIABP	Data Type	Length	Olere Dielle							
BMI BODYFAT BSA DIABP			Sign.Digits	Origin	Comment	Description	def:Display	Method	CodeList	WhereClaus
BODYFAT BSA DIABP										WC.IT.BMI
BSA DIABP										WC.IT.BODYFAT
DIABP										WC.IT.BSA
										WC.IT.DIABP
RMSIZE										WC.IT.FRMSIZE
IEIGHT										WC.IT.HEIGHT
IR										WC.IT.HR
IAP										WC.IT.MAP
PULSE										WC.IT.PULSE
RESP										WC.IT.RESP
SYSBP										WC.IT.SYSBP
EMP										WC IT TEMP
VEIGHT										WC IT WEIGHT
		Ins	ert row							Rem
		1112								Kenn
	AFUULSE ESP YSBP EMP FEIGHT	AF ULSE ESP YSBP EMP EIGHT	AF UUSE UUSE SP	Ar ULSE	Ar	Arr	Arr	Arr	Arr	Arr Image: Constraint of the second sec

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 unnessary² rows(using the "Remove row" button), we can start adding the data types:

² This depends of course on your own study.

CL.VSTESTO	D - Vital Sign	ns Test Coo	le				
New OID:							
OID	Name	Data Ty	pe	Length	Sign.Digits	Origin	Commer
IT.DIABP	DIABP	integer					
IT.FRMSIZE	FRMSIZE		-				
IT.HEIGHT	HEIGHT	integer					-
IT.SYSBP	SYSBP	float					
IT.WEIGHT	WEIGHT	text					-
		date	=				
		partialda	t				
		time					
		nartialtin					
		datatima	-				
		uatetime					

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

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.

CL.VSTEST	CD - Vital Sigi	ns Test Code								
New OID:								VL.VSORRE	s	
OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:Display	Method	CodeLis
IT.DIABP	DIABP	integer	3							
IT.FRMSIZE	FRMSIZE	text	6							
IT.HEIGHT	HEIGHT	float	5	1				1		
IT.SYSBP	SYSBP	integer	3							
IT.WEIGHT	WEIGHT	float	5	1						

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 cells "Description"³ are colored red, meaning we do need provide values there:

^{3 &}quot;Description" replaces "def:Label" in define.xml v.2.0

🛓 Create	new SDTM V	alueList from	existing CodeL	ist				
?								
	CL.VSTESTC	CD - Vital Sigr	ns Test Code					
	New OID:							VL.VSORRES
	010	blassa	Data Tura	Land	Oise Disite	Orinin	Ormeret	Description
	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 source 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:

Designing	/Updating Origin for Item: IT.DIABP	٢
?	Origin type: Assigned	
	 Protocol Data that is determined by individual judgment Derived (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: OK Cancel	
	OK Cancel	

where can select between:

- 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:

Designing	g/Updating Origin for Item: IT.DIABP	x
?	Origin type: 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	
	⊇⊇ ⊇ Page range: first page - last page	
	First page:	
	Last page:	
	OK Cancel	

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			Γ
IT.HEIGHT	HEIGHT	float	5	1		Γ
IT.SYSBP	SYSBP	integer	3			ſ
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", leading to:



and scroll down 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:

Comme	nt:		
	External document for com	ment	
		Number of RangeCheck	ks: 1
		SUBJID	
		USUBJID	
		CE.CESEQ	
(Comparator: EQ 🛛 🔽 Item OID:	CE.CEGRPID	CheckValue:
		CE.CEREFID	
		CE.CESPID	
		CE.CETERM	
		CE.CEDECOD	

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

OID:		WC.IT.	DIABP	
Com	ment:			
	External document for comm	ient		
		Number of RangeCheck	s• 1	
		Number of Nangeeneer		
		TS.TSVAL		
		TS.TSVAL VS.VSSEQ	-	
		TS.TSVAL VS.VSSEQ VS.VSGRPID	-	
		TS.TSVAL VS.VSSEQ VS.VSGRPID VS.VSSPID		
		TS.TSVAL VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD		
	Comparator: EQ 🔽 Item OID:	TS.TSVAL VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD VS.VSTEST	CheckValue: DIA	BP
	Comparator: EQ 🔽 Item OID:	TS.TSVAL VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD VS.VSTEST VS.VSCAT	CheckValue: DIA	BP
	Comparator: EQ 🔽 Item OID:	TS.TSVAL VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD VS.VSTEST VS.VSCAT VS.VSSCAT	CheckValue: DIA	BP
	Comparator: EQ 🔽 Item OID:	TS.TSVAL VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD VS.VSTEST VS.VSCAT VS.VSSCAT VS.VSPOS	CheckValue: DIA	BP

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

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 futher) 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 do	ocument	x
i	No external document	
	LEAF.SUPPDOC1	
	LEAF.SUPPDOC2	
	O Page list (physical reference)	
	Named destinations	
	Page list / List of named destinations	
	#DIABP	
	Page range: first page - last page	
	First page:	
	Last page:	
	ОК	

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":

OID:	VS.VSORRES
Name:	VSORRES
Data type:	text
Current Length:	80
New Length:	
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 comme	ent
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 descript	ion:
current ValueList OID:	NO VALUELIST ASSIGNED

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
	VL.CL.VSTESTCD
	VL.VSORRES
	NO VALUELIST

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

```
<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" 0ID="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 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 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 in "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":

CL.LBIESICD - Laboratory Test Code	
CL.LBTEST - Laboratory Test Name	
CL.UNITS - Units	
CL.FRQ - Frequency	
CL.EGTEST - ECG Test Name	=
CL.EGTESTCD - ECG Test Codes	
CL.EGPOS - ECG Position of Subject	
CL.EGSTRESC - ECG Result	
CL.EGMETHOD - Method of ECG Test	
MyStudy:CL.DRUGTRTF - Assigned Study Drug	•
OK Cancel	

This time we select the "units" codelist, which is then transformed to a ValueList:

CL UNITE	Unito										
CL.UNITS -	Units										
New OID:								VL.CL.UN	NITS		
OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:Display	Method	CodeList	Wh
IT.mol	mol										WC
IT.dmol	dmol										WC
IT.cmol	cmol										WC
IT.mmol	mmol										WC
IT.umol	umol										WC
IT.nmol	nmol										WC
IT.pmol	pmol										WC
IT.fmol	fmol										WC
IT.amol_	amol										WC
IT.eq	eq										WC
IT.meq	meq										WC
IT.m3	m3										WC
IT.ft3	ft3										WC
IT.deg	deg										WC
IT.cm2	cm2										WC
IT mm2	mm2										WC

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

New OID:	New OID:								VL.VSORRESU		
		1		1		1					
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	g/Updating WhereClause for Item: IT.mmHg		×
?	OID:	WC.IT.mmH	g
	Comment:	mmHg for di	iastolic and blood pressure
	External document for comm	nent	
		Number of RangeChecks:	1 -
			<u> </u>
		DOMAIN	
		STUDYID SUBJID USUBJID	
	Comparator: EQ 🔽 Item OID:	CE.CESEQ CE.CEGRPID	CheckValue:
		CE.CESPID	
		CE.CETERM	

We cannnot 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:

	DOMAIN STUDYID SUBJID			Add to or remov	e from list
	USUBJID			Add to list	Remove f
	CE.CESEQ				
Comparator: IN The Item OID:	CE.CEGRPID		CheckValues:		
	CE.CEREFID				
	CE.CESPID				
	CE.CETERM				
	CE.CEDECOD				
	CE.CECAT	•			

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:

Comparator: IN 💌 Item OID:	TS.TSSEQ TS.TSGRPID TS.TSPARMCD TS.TSPARM TS.TSVAL VS.VSSEQ VS.VSSEQ VS.VSGRPID VS.VSSPID		CheckValues:	Add to or remov Add to list SYSBP DIABP	e from list Remove f
	VS.VSSPID VS.VSTESTCD	ł			
	VS.VSTEST				
	VS.VSCAT	•]		

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 Cod	leList definiti	ons from O	DM into de	efine.xml				
Selected CodeList definitions from ODM into define.xml								
CodeList definitions from File into define.xml								
Create	new SDTM C	odeList fro	om existin	a CodeList				
Create	e new SDTM C	odeList fro	om Measu	rementUnits				

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

🛃 Create new SDTM Code	List from existing CodeList		×
?	CL.VSTESTCD - Vital Signs Test Code		_
	CL.UNITS - Units		A
	CL.FRQ - Frequency		
	CL.EGTEST - ECG Test Name		
	CL.EGTESTCD - ECG Test Codes		
	CL.EGPOS - ECG Position of Subject		
	CL.EGSTRESC - ECG Result		
	CL.EGMETHOD - Method of ECG Test		
	MyStudy:CL.DRUGTRTF - Assigned Study	Drug	•
	Insert row	Remove row	

which then displays the full list of units available⁴:

⁴ Unfortunately the CDISC-CT develops one huge list of units without any system, and independent of the kind of test. We hope that CDISC will however soon switch to <u>UCUM</u> for units.

Create new SDTM CodeList from existing CodeList			
-			
CL UNITS - Units			-
CLOWING - ONICO			
New OID:			
New Name:			
New DataType:	text		-
New SASFormatName:			
Remove rows until only	those remain the	at you want to appear in the Codel ist	
You can also add new re	OWS.	at you main to uppour in the Couclist	
Insert ro	w	Remove row	
CodedValue	Language	Decode	
mol			-
cmol			=
mmol			
umol			
nmol			
pmol			
fmol			
amol			
eq			
meg			
m3			
ft3			
deg			
cm2			
mm2			
ft2			
in2			
ELISA unit			
ELISA unit/dose			
ELISA unit/ml			

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:

CL.UNITS - Units			-			
New OID:	CL.BLOODPRESSU	REUNITS				
New Name:	Blood pressure unit:	Blood pressure units				
New DataType:	text		-			
New SASFormatNa	ame:					
New SASFormatNa Remove rows until You can also add r	ame: only those remain that yo ew rows.	u want to appear in the Cod	leList.			
New SASFormatNa Remove rows until You can also add r Inse	ame: only those remain that yo ew rows. ert row	w want to appear in the Cod Remove row	leList.			

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

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

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):

Edit SDTM ValueList	ten her generige										
VL.VSORRESU											
	Name Weight Linits	Data Type	Length	Sig	Origin	Com	Description	def	Meth	CodeList	WhereClau.
IT.WEIGHT.USA	Weight Units	text	2		Protocol		Weight: original units				WCITLB
IT.BLOODPRESSURE	Blood pressure Units	text	4		Protocol		Blood pressure: orig			CL.VSTESTCD	WC.IT.mm
										CL.LOC CL.VESNONULL CL.BESTCD CL.BEST CL.BLOODPRESSUREUNITS CL.FRQ CLFRQ CLEGTEST Blood pressure ur Blood pressure ur Blood pressure ur	

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 names of the OIDs do not care, 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.:

🚣 Create new SDTM	A CodeList from existing CodeList				X				
?	CL.UNITS - Units	CL.UNITS - Units							
	New OID:	New OID: CL.WEIGHTEUROPE							
New Name:		Weight units for Eur							
	New DataType:	text	-						
	New SASFormatName:								
	Remove rows until only those remain that you want to appear in the CodeList.								
	You can also add new								
	Insert re	ow	v Remove row						
	CodedValue	Language	Decode						
	kg								

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

<u> </u>	CL.UNITS - Units	CL.UNITS - Units							
	New OID:	CL.WEIGHTUSA							
New Name: New DataType:	New Name:	Weight units for the	Weight units for the USA						
	text	text							
	New SASFormatNar	ne:							
	Remove rows until only those remain that you want to appear in the CodeList.								
	You can also add ne								
	Inser	trow	Remove row						
	CodedValue	Language	Decode						
	IB								

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		WhereC
IT.WEIGHT.EUROPE	Weight Units	text	2		Protocol		Weight: ori	1		CL.WEIGHTEUROF	PE	WC.IT.kg
IT.BLOODPRESSURE	Blood pres	text	4		Protocol		Blood pres			CL.BLOODPRESSU CL.WEIGHTEUROPI CL.WEIGHTUSA CL.UNITS		WC.IT.m

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:

OID:			WC.IT.V	VEI	GHT.EUROPE		
Comment:	omment:			expressed in kg for VSTESTCD=WEIGHT and country=GEF			
	External document for	comment					
		Numbe	er of RangeChecks	: [2 +		
	Comparator: EQ	▼ Item OID:	SV.SVENDY TS.TSSEQ TS.TSGRPID TS.TSPARMCD TS.TSPARM TS.TSVAL VS.VSSEQ VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD VS.VSTEST		C C	heckValue: WEIGHT	
Comparato	r: IN 💌 Item OID:	DM.BRTHDTC DM.AGE DM.AGEU DM.SEX DM.RACE DM.ETHNIC DM.ARMCD DM.ARM DM.COUNTRY DM.DMDTC			CheckValues:	Add to or remove from list Add to list Remov GER FRA	

having two checks:

- VSTESTCD=WEIGHT and:
- DM.COUNTRY has either the value "GER" (Germany) or "FRA" (France).

Also remark that we changed the OID for the "WhereClause" to make it more clear (although essentially it doesn't matter).

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

OID:	WC.IT	T.WEIGHT.USA		
Comment:	ıt is e	It is expressed in LB for VSTESTCD=WEIGHT and country		
External document for comment				
	Number of RangeChec	ks: 2 -		
Comparator: EQ 💌 Item OID:	SV.SVENDY TS.TSSEQ TS.TSGRPID TS.TSPARMCD TS.TSPARM TS.TSVAL VS.VSSEQ VS.VSSEQ VS.VSGRPID VS.VSSPID VS.VSTESTCD VS.VSTEST	CheckValue: WEIGHT		
Comparator: EQ 🔽 Item OID:	DM.INVNAM DM.BRTHDTC DM.AGE DM.AGEU DM.SEX DM.RACE DM.ETHNIC DM.ARMCD DM.ARM DM.COUNTRY	CheckValue: USA		

defining two conditions:

```
    VSTESTCD=WEIGHT
and:
DM.COUNTRY=USA
```

```
<def:ValueListDef OID="VL.VSORRESU">
   <ItemRef Item0ID="IT.WEIGHT.EUROPE" Mandatory="No" OrderNumber="1">
      <def:WhereClauseRef WhereClauseOID="WC.IT.WEIGHT.EUROPE"/>
   </ItemRef>
   <ItemRef Item0ID="IT.WEIGHT.USA" Mandatory="No" OrderNumber="2">
      <def:WhereClauseRef WhereClauseOID="WC.IT.WEIGHT.USA"/>
   </ItemRef>
   <ItemRef Item0ID="IT.BL00DPRESSURE" 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 0ID="WC.IT.WEIGHT.EUR0PE" def:Comment0ID="COM.WC.IT.WEIGHT.EUR0PE">

<RangeCheck Comparator="EQ" SoftHard="Soft" def:Item0ID="VS.VSTESTCD">

<CheckValue>WEIGHT</CheckValue>

</RangeCheck>

<RangeCheck Comparator="IN" SoftHard="Soft" def:Item0ID="DM.COUNTRY">

<CheckValue>GER</CheckValue>

<CheckValue>GER</CheckValue>

</RangeCheck>

</RangeCheck>

</RangeCheck>
```

and the corresponding comment:

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

erties 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
	NU OL MOTEOTOD

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	
	VL.CL.VSTESTCD	
	VL.VSORRES	
	VL.VSORRESU	
	NO VALUELIST	

which appears in the define.xml as:

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

Value Level Metadata

Value Level Metadata - Vital Signs [VSORRES]

Variable	Where	Туре	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"] < <u>Frame size</u> >	CRF Page <u>25</u>	
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 <u>25</u>	

Value Level Metadata - Vital Signs [VSORRESU]

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

This finalizes our tutorial about the use of the "Where Clause". Once again, when using "Where Clause" you do not need nested value levels anymore, and in define.xml 2.0, you should not use nested valuelevel metadata descriptions.

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 XML editing.