

SM03: VBA for CRT

Automating the Master Spec

Name: Matt Methereil

Title: Senior Programmer

Company: PHASTAR

Process Overview

1. P21 generates a blank template spec, based on XPT files.
2. We fill in the template, making it into the master spec
3. P21 converts this master spec into the define.xml file

Template spec: Study tab

The first tab is called Study, and requires use to enter the study name and description as well as the protocol name.

A	B
Attribute	Value
StudyName	
StudyDescription	
ProtocolName	
StandardName	SDTM-IG
StandardVersion	3.2
Language	en

Master spec: Study tab

These are all mapped to the define.xml, although only the study name is visible in the version we see in our browser.

Attribute	Value
StudyName	PhUSEExample
StudyDescription	A phase 1, multicenter, prospective, randomized, double blind, placebo-controlled study to reduce incidence of disease with PhUSEExample
ProtocolName	ABC12345
StandardName	SDTM-IG
StandardVersion	3.2
Language	en

i:\VBA for CRT\Study area\PhUSEExample-2019-08-10T13-42.xml	
Study: PhUSEExample Data D...	

Tabulation Datasets for Study **PhUSEExample** (SDTM-IG 3.2)

Dataset	Description	Class	Structure	Purpose	Keys
TA	Trial Arms	TRIAL DESIGN	One record per arm per element	Tabulation	STUDYID, ARMCD, TAETORD
TE	Trial Elements	TRIAL DESIGN	One record per arm per element	Tabulation	STUDYID, ETCD

```
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns="http://www.cdisc.org/ns/odm/v1.3"
xmlns:def="http://www.cdisc.org/ns/def/v2.0"
ODMVersion="1.3.2"
FileType="Snapshot"
FileOID="PhUSEExample.SDTM-IG.3.2"
CreationDateTime="2019-08-10T13:42:12">
<Study OID="PhUSEExample.SDTM-IG.3.2">
  <GlobalVariables>
    <StudyName>PhUSEExample</StudyName>
    <StudyDescription>A phase 1, multicenter,
    placebo-controlled study to reduce incidence of disease with PhUSEExample</StudyDescription>
    <ProtocolName>ABC12345</ProtocolName>
  </GlobalVariables>
```

Predecessors

Predecessor in ValueLevel tab (ADaM)

When entering a Predecessor in the ValueLevel tab in v2.x, it will not be visible when viewing in a browser even though it is present in the XML data.

Variable	Where Clause	Data Type	Length	Significant Digit	Format	Mandatory	Codelist	Origin	Pages	Method	Predecessor	Value Level Cor	Join Comment
AVALC	ADLB.PARAMCD.EQ.HYS1FL	text	200			No		Derived		ADLB.AVALC.HYS1FL			
AVALC	ADLB.PARAMCD.EQ.HYS2FL	text	200			No		Derived		ADLB.AVALC.HYS2FL			
AVALC	ADLB.PARAMCD.NOTIN	text	200			No		Predecessor			LB.LBSTRESC		

```
<ItemDef OID="IT.ADLB.AVALC.ADLB.PARAMCD.NOTIN.HYS1FL-c6f507ff"
  Name="AVALC.ADLB.PARAMCD.NOTIN.HYS1FL-c6f507ff"
  DataType="text"
  Length="200"
  SASFieldName="AVALC">
  <def:Origin Type="Predecessor">
    <Description>
      <TranslatedText xml:lang="en">LB.LBSTRESC</TranslatedText>
    </Description>
  </def:Origin>
</ItemDef>
```

Variable	Where	Type	Length / Display Format	Controlled Terms or Format	Origin	Derivation/Comment
AVALC	PARAMCD EQ HYS1FL (AST or ALT elevation)	text	200		Derived	For subjects in safety analysis set (ADSL.SAFFL=Y), equal to Y if patient has AST or ALT elevation (PARAMCD in ('L02101U1','L02102U1') and CRIT1FL='Y') between treatment start and treatment end +30 days (ADSL.TRSTD<=ADT<ADSL.TRTEDT+30), otherwise equal to N.
AVALC	PARAMCD EQ HYS2FL (AT or Total Bilirubin (THL) elevation)	text	200		Derived	For subjects in safety analysis set (ADSL.SAFFL=Y), equal to Y if patient has (ALT > 3X ULN or AST > 3X ULN) and (TBL >2X ULN within 14 days on or after AT elevation): PARAMCD='L02110U1' and CRIT1FL='Y' and TRTSDT<=ADT<TRTEDT+30, within 14 days of AT elevation (see HYS1FL).
AVALC	PARAMCD NOTIN ('HYS1FL' (AST or ALT elevation), 'HYS2FL' (AT or Total Bilirubin (THL) elevation))	text	200		Predecessor	

Predecessor in ValueLevel tab (ADaM)

To account for this we must add a comment. On the next slide is a macro that will copy the value into a comment for all rows that don't currently have a comment.

Order	Dataset	Variable	Where Clause	Data Type	Length	Significant Dig	Format	Mandator	Codelist	Origin	Pages	Method	Predecessor	Value Level Com
	ADLB	AVALC	ADLB.PARAMCD.EQ.HY	text	200			No		Derived		ADLB.AVALC.HYS1FL		
	ADLB	AVALC	ADLB.PARAMCD.EQ.HY	text	200			No		Derived		ADLB.AVALC.HYS2FL		
	ADLB	AVALC	ADLB.PARAMCD.NOTIN	text	200			No		Predecessor			LB.LBSTRESC	ADLB.LB.LBSTRESC

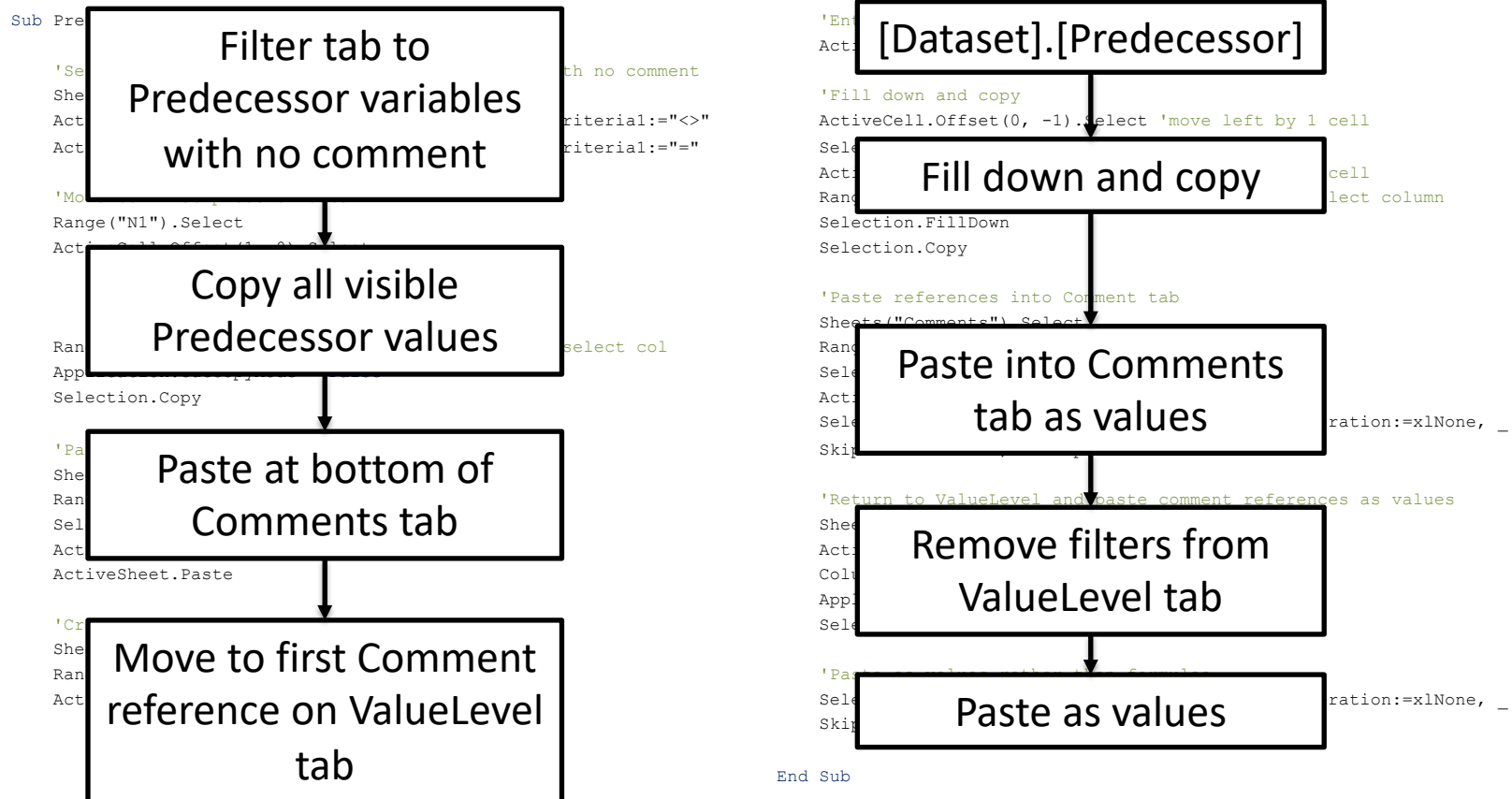
```

<ItemDef OID="IT.ADLB.AVALC.ADLB.PARAMCD.NOTIN.HYS1FL-c6f507ff"
  Name="AVALC.ADLB.PARAMCD.NOTIN.HYS1FL-c6f507ff"
  DataType="text"
  Length="200"
  SASFieldName="AVALC"
  def:CommentOID="COM.ADLB.LB.LBSTRESC">
<def:Origin Type="Predecessor">
  <Description>
    <TranslatedText xml:lang="en">LB.LBSTRESC</TranslatedText>
  </Description>
</def:Origin>
</ItemDef>

```

Variable	Where	Type	Length / Display Format	Controlled Terms or Format	Origin	Derivation/Comment
AVALC	PARAMCD EQ HYS1FL (AST or ALT elevation)	text	200		Derived	For subjects in safety analysis set (ADSL.SAFFL=Y), equal to Y if patient has AST or ALT elevation (PARAMCD in ('L02101U1','L02102U1') and CRIT1FL=Y) between treatment start and treatment end +30 days (ADSL.TRTSDT<=ADT<ADSL.TRTEDT+30), otherwise equal to N.
AVALC	PARAMCD EQ HYS2FL (AT or Total Bilirubin (THL) elevation)	text	200		Derived	For subjects in safety analysis set (ADSL.SAFFL=Y), equal to Y if patient has (ALT > 3X ULN or AST > 3X ULN) and (TBL >2X ULN within 14 days on or after AT elevation): PARAMCD='L02110U1' and CRIT1FL='Y' and TRTSDT<=ADT<TRTEDT+30, within 14 days of AT elevation (see HYS1FL).
AVALC	PARAMCD NOTIN ("HYS1FL" (AST or ALT elevation) , "HYS2FL" (AT or Total Bilirubin (THL) elevation))	text	200		Predecessor	LB.LBSTRESC

Predecessor in ValueLevel tab (ADaM)



Predecessor in ValueLevel tab (ADaM)

Alternatively, you can simply use the stylesheet that comes with version 3 of P21 Community to display the Predecessor without having a comment in the master spec.

Variable	Where Clause	Data Type	Length	Significant Digit	Format	Mandatory	Codelist	Origin	Pages	Method	Predecessor	Value Level Cor	Join Comment
AVALC	ADLB.PARAMCD.EQ.HYS	text	200			No		Derived		ADLB.AVALC.HYS1FL			
AVALC	ADLB.PARAMCD.EQ.HYS	text	200			No		Derived		ADLB.AVALC.HYS2FL			
AVALC	ADLB.PARAMCD.NOTIN	text	200			No		Predecessor			LB.LBSTRESC		

```
<ItemDef OID="IT.ADLB.AVALC.ADLB.PARAMCD.NOTIN.HYS1FL-c6f507ff"
  Name="AVALC.ADLB.PARAMCD.NOTIN.HYS1FL-c6f507ff"
  DataType="text"
  Length="200"
  SASFieldName="AVALC">
  <def:Origin Type="Predecessor">
    <Description>
      <TranslatedText xml:lang="en">LB.LBSTRESC</TranslatedText>
    </Description>
  </def:Origin>
</ItemDef>
```

PARAMCD	Parameter Code	text	Analysis Parameter	8	Parameter Code (ADLB) [33 Terms]	See value-level metadata for variable definitions.
PARAMCD = "HYS1FL" (AST or ALT elevation)		text		8		Assigned Assigned to "HYS1FL"
PARAMCD = "HYS2FL" (AT or Total Bilirubin (THL) elevation)		text		8		Assigned Assigned to "HYS2FL"
PARAMCD NOT IN ("HYS1FL" (AST or ALT elevation), "HYS2FL" (AT or Total Bilirubin (THL) elevation))		text		8	Parameter Code (ADLB) [33 Terms]	Predecessor: SUPPLB.QVAL where SUPPLB.QNAM="AZLABCD"

ValueLevel in P21 Community v3

In versions 3.0.0 and 3.0.1 of P21 Community Edition, there is a bug in the define generator that means ValueLevel entries display all derivations, comments and predecessors for all ValueLevel entries that use the same where clause. In this example, there are four entries, for AVALC, PARAM, PARAMREP and DTYPE:

PARAM VLM		Parameter	text	Analysis Parameter	55	Parameter (ADLB) [33 Terms]	See value-level metadata for variable definitions.
	PARAMCD = "HYS1FL" (AST or ALT elevation)		text		200		Derived Assigned Assigned Assigned
	PARAMCD = "HYS2FL" (AT or Total Bilirubin (THL) elevation)		text		200		Derived Assigned Assigned Assigned
	PARAMCD NOT IN ("HYS1FL" (AST or ALT elevation), "HYS2FL" (AT or Total Bilirubin (THL) elevation))		text		200	Parameter (ADLB) [66 Terms]	Predecessor: LB.LBSTRESC Predecessor: SUPPLB.QVAL where SUPPLB.QNAM='AZLABCD' Predecessor: SUPPLB.QVAL where SUPPLB.QNAM='AZLABNM' Predecessor: LB.LBTEST

ValueLevel in P21 Community v3

In version 3.0.1 of P21 Community Edition, a valuelevel item is defined by the ValueLevel dataset and the where clause definition. Specifically:

IT.[VL.dataset].[WC.dataset].[WC.variable].[WC.comparator].[WC.value]

ID	Dataset	Variable	Comparator	Value
ADLB.PARAM.1	ADLB	PARAMCD	EQ	HYS1FL
ADLB.PARAM.2	ADLB	PARAMCD	EQ	HYS2FL
ADLB.PARAM.3	ADLB	PARAMCD	NOTIN	HYS1FL,HYS2FL

Dataset	Variable	Where Clause
ADLB	PARAM	ADLB.PARAM.1
ADLB	PARAM	ADLB.PARAM.2
ADLB	PARAM	ADLB.PARAM.3

```
<def:ValueListDef OID="VL.ADLB.PARAM">
  <ItemRef ItemOID="IT.ADLB.ADLB.PARAMCD.EQ" bc0e270c2a2edd403638a66d49a53b447fcfdf1"
    OrderNumber="1"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADLB.PARAM.1"/>
  </ItemRef>
  <ItemRef ItemOID="IT.ADLB.ADLB.PARAMCD.EQ" c52dcfab08360189769441c49fe03549004e13"
    OrderNumber="2"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADLB.PARAM.2"/>
  </ItemRef>
  <ItemRef ItemOID="IT.ADLB.ADLB.PARAMCD.NOTIN" f9920f40a2781b73c437ba590ae73ded63aa53e"
    OrderNumber="3"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADLB.PARAM.3"/>
  </ItemRef>
</def:ValueListDef>
```

So the ValueLevel item is only unique if the where clause is unique within the dataset

ValueLevel in P21 Community v3

If multiple variables in a dataset have the same where clause, this results in the define.xml displaying all sources and codelists associated with the where clause:

AVALC	PARAMCD = "HYS1FL" (AST or ALT elevation)	Analysis Value (C)	text	10	See value-level metadata for variable definitions.
	PARAMCD = "HYS2FL" (AT or Total Bilirubin (THL) elevation)		text	200	Derived Assigned Assigned Assigned For subjects in safety analysis set (ADSL.SAFL=Y), equal to Y if patient has AST or ALT elevation (PARAMCD in 'L02101U1', 'L02102U1') and CRIT1FL=Y) between treatment start and treatment end +30 days (ADSL.TRTSDT<=ADT<ADSL.TRTEDT+30), otherwise equal to N. Assigned to "AST or ALT elevation"
	PARAMCD NOT IN ("HYS1FL" (AST or ALT elevation), "HYS2FL" (AT or Total Bilirubin (THL) elevation))		text	200	Derived Assigned Assigned Assigned For subjects in safety analysis set (ADSL.SAFL=Y), equal to Y if patient has (ALT > 3X ULN or AST > 3X ULN) and (TBL > 2X ULN within 14 days on or after AT elevation): PARAMCD='L02110U1' and CRIT1FL=Y and TRTSDT<=ADT<TRTEDT+30, within 14 days of AT elevation (see HYS1FL). Assigned to "AT or Total Bilirubin (THL) elevation"
				200	Predecessor: LB.LBSTRESC Predecessor: SUPPLB.QVAL where SUPPLB.QNAM='AZLABCD' Predecessor: SUPPLB.QVAL where SUPPLB.QNAM='AZLABNM' Predecessor: LB.LBTEST LB.LBSTRESC

Dataset	Variable	Where Clause	Data Type	Length	Codelist	Origin	Method	Predecessor	Comment
ADLB	AVALC	ADLB.PARAMCD.EQ.HYS1FL	text	200		Derived	ADLB.AVALC.HYS1FL		
ADLB	AVALC	ADLB.PARAMCD.EQ.HYS2FL	text	200		Derived	ADLB.AVALC.HYS2FL		
ADLB	AVALC	ADLB.PARAMCD.NOTIN.HYS1FL,HYS2FL	text	200		Predecessor		LB.LBSTRESC	ADLB.LB.LBSTRESC
ADLB	PARAMCD	ADLB.PARAMCD.EQ.HYS1FL	text	8		Assigned			ADLB.PARAMCD.HYS1FL
ADLB	PARAMCD	ADLB.PARAMCD.EQ.HYS2FL	text	8		Assigned			ADLB.PARAMCD.HYS2FL
ADLB	PARAMCD	ADLB.PARAMCD.NOTIN.HYS1FL,HYS2FL	text	8	ADLB.PARAMCD	Predecessor		SUPPLB.QVAL where SUPPLB.QNAM='AZLABCD'	ADLB.SUPPLB.QVAL where SUPPLB.QNAM='AZLABCD'
ADLB	PARAM	ADLB.PARAMCD.EQ.HYS1FL	text	200		Assigned			ADLB.HY1.PARAMREP
ADLB	PARAM	ADLB.PARAMCD.EQ.HYS2FL	text	200		Assigned			ADLB.HY2PARAMREP
ADLB	PARAM	ADLB.PARAMCD.NOTIN.HYS1FL,HYS2FL	text	200	ADLB.PARAM	Predecessor		SUPPLB.QVAL where SUPPLB.QNAM='AZLABNM'	ADLB.SUPPLB.QVAL where SUPPLB.QNAM='AZLABNM'
ADLB	PARAMREP	ADLB.PARAMCD.EQ.HYS1FL	text	40		Assigned			ADLB.HY1.PARAMREP
ADLB	PARAMREP	ADLB.PARAMCD.EQ.HYS2FL	text	40		Assigned			ADLB.HY2PARAMREP
ADLB	PARAMREP	ADLB.PARAMCD.NOTIN.HYS1FL,HYS2FL	text	40		Predecessor		LB.LBTEST	ADLB.LB.LBTEST

ValueLevel in P21 Community v2

In version 2 of P21 Community Edition, the where clause definition is used in conjunction with the ValueLevel dataset and variable name to create a unique ValueLevel ID of the form:

IT.[VL.dataset].[VL.variable].[WC.dataset].[WC.variable].[WC.comparator].[WC.value]

ID	Dataset	Variable	Comparator	Value
ADLB.PARAM.1	ADLB	PARAMCD	EQ	HYS1FL
ADLB.PARAM.2	ADLB	PARAMCD	EQ	HYS2FL
ADLB.PARAM.3	ADLB	PARAMCD	NOTIN	HYS1FL,HYS2FL

Study Datasets Variables ValueLevel WhereClauses Cc ... + -

Dataset	Variable	Where Clause
ADLB	PARAM	ADLB.PARAM.1
ADLB	PARAM	ADLB.PARAM.2
ADLB	PARAM	ADLB.PARAM.3

dy Datasets Variables ValueLevel WhereClauses Cc ... + -

```
<def:ValueListDef OID="VL_ADLB.PARAM">
  <ItemRef ItemOID="IT_ADLB.PARAM_ADLB.PARAMCD_EQ_HYS1FL-b57d7cc8" OrderNumber="1"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC_ADLB.PARAMCD_EQ_HYS1FL-b57d7cc8"/>
  </ItemRef>
  <ItemRef ItemOID="IT_ADLB.PARAM_ADLB.PARAMCD_EQ_HYS2FL-b4f654dc" OrderNumber="2"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC_ADLB.PARAMCD_EQ_HYS2FL-b4f654dc"/>
  </ItemRef>
  <ItemRef ItemOID="IT_ADLB.PARAM_ADLB.PARAMCD_NOTIN_HYS1FL-c6f507ff" OrderNumber="3"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC_ADLB.PARAMCD_NOTIN_HYS1FL-c6f507ff"/>
  </ItemRef>
</def:ValueListDef>
```

ValueLevel in P21 Community v2

This uniqueness means we don't get repeated sources in version 2.2.

P21 fixed this by adding the ValueLevel variable name back into the ItemOID for version 3.0.2.

This brings us on to another problem experienced in version 2.2: composite where clauses.

Where Clauses

Composite Where Clauses in P21 v2

We can combine multiple where clauses with an AND operator if we give them all the same ID on the where clause tab.

To achieve this, v2.2 concatenates all the where clauses in the ItemOID, as well as in the WhereClauseOID.

ID	Dataset	Variable	Comparator	Value
ADSK.DTYPE.2	ADSK	PARAMCD	EQ	K
ADSK.DTYPE.2	ADSK	AVISITN	IN	4,7,9,10,11,12,13,14,15
ADSK.DTYPE.2	ADSK	ATPTN	EQ	PRE-DIALYSIS

```
<ItemRef ItemOID="IT.ADSK.DTYPE.ADSK.PARAMCD.EQ.K.ADSK.AVISITN.IN.4.ADSK.ATPTN.EQ.PRE-DIALYSIS-1daf034b"
  OrderNumber="2"
  Mandatory="No"
  MethodOID="MT.ADSK.DTYPE.ADJUSTAT">
  <def:WhereClauseRef WhereClauseOID="WC.ADSK.PARAMCD.EQ.K.ADSK.AVISITN.IN.4.ADSK.ATPTN.EQ.PRE-DIALYSIS-1daf034b"/>
</ItemRef>
```

Variable	Where
DTYPE	PARAMCD EQ K (Potassium (mmol/L))
DTYPE	PARAMCD EQ K (Potassium (mmol/L)) AND AVISITN IN ("4", "7", "9", "10", "11", "12", "13", "14", "15") AND ATPTN EQ PRE-DIALYSIS

Duplicate Where Clauses in P21 v2

Because the WhereClauseOID in version 2.2 is created from the definition of the where clause, and not its user-defined ID, this means we cannot have multiple instances of the same where clause in the spec, or they will all appear each time one of them is expected.

Dataset	Variable	Where Clause
ADSK	STUDYID	ADSK.STUDYID.1
ADSK	STUDYID	ADSK.STUDYID.2
ADSK	USUBJID	ADSK.USUBJID.1
ADSK	USUBJID	ADSK.USUBJID.2

ID	Dataset	Variable	Comparator	Value
ADSK.STUDYID.1	ADSK	PARAMCD	EQ	DIALKC
ADSK.STUDYID.2	ADSK	PARAMCD	NE	DIALKC
ADSK.USUBJID.1	ADSK	PARAMCD	EQ	DIALKC
ADSK.USUBJID.2	ADSK	PARAMCD	NE	DIALKC

```
<def:ValueListDef OID="VL.ADSK.STUDYID">
  <ItemRef ItemOID="IT.ADSK.STUDYID.ADSK.PARAMCD.EQ.DIALKC-02a44f0e" OrderNumber="1"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.PARAMCD.EQ.DIALKC-02a44f0e"/>
  </ItemRef>
  <ItemRef ItemOID="IT.ADSK.STUDYID.ADSK.PARAMCD.NE.DIALKC-cc537c8c" OrderNumber="2"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.PARAMCD.NE.DIALKC-cc537c8c"/>
  </ItemRef>
</def:ValueListDef>
<def:ValueListDef OID="VL.ADSK.USUBJID">
  <ItemRef ItemOID="IT.ADSK.USUBJID.ADSK.PARAMCD.EQ.DIALKC-02a44f0e" OrderNumber="1"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.PARAMCD.EQ.DIALKC-02a44f0e"/>
  </ItemRef>
  <ItemRef ItemOID="IT.ADSK.USUBJID.ADSK.PARAMCD.NE.DIALKC-cc537c8c" OrderNumber="2"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.PARAMCD.NE.DIALKC-cc537c8c"/>
  </ItemRef>
</def:ValueListDef>
```

```
<def:WhereClauseDef OID="WC.ADSK.PARAMCD.EQ.DIALKC-02a44f0e">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="EQ">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.ADSK.PARAMCD.NE.DIALKC-cc537c8c">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="NE">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.ADSK.PARAMCD.EQ.DIALKC-02a44f0e">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="EQ">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.ADSK.PARAMCD.NE.DIALKC-cc537c8c">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="NE">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
```

Duplicate Where Clauses in P21 v2

Which looks like this:

Parameter Value List - ADSK [STUDYID]

Variable	Where	Type	Length / Display Format	Controlled Terms or Format	Origin	Derivation/Comment
STUDYID	PARAMCD EQ DIALKC (Dialysate Potassium Concentration (mmol/L)) PARAMCD EQ DIALKC (Dialysate Potassium Concentration (mmol/L))	text	11		Predecessor	FA.STUDYID
STUDYID	PARAMCD NE DIALKC PARAMCD NE DIALKC	text	11		Predecessor	LB.STUDYID

Parameter Value List - ADSK [USUBJID]

Variable	Where	Type	Length / Display Format	Controlled Terms or Format	Origin	Derivation/Comment
USUBJID	PARAMCD EQ DIALKC (Dialysate Potassium Concentration (mmol/L)) PARAMCD EQ DIALKC (Dialysate Potassium Concentration (mmol/L))	text	30		Predecessor	FA.USUBJID
USUBJID	PARAMCD NE DIALKC PARAMCD NE DIALKC	text	30		Predecessor	LB.USUBJID

De-duplicating Where Clauses in P21 v2

To combat this, we must de-duplicate the WhereClause tab when all columns are equal, including the ID.

While our user-defined Where Clause ID does not show up in the define, it is used to connect the two tabs during the define creation process.

If we set this ID to be equal to [dataset].[variable].[comparator].[value], then we can use the same on both the ValueLevel and WhereClauses tabs without having to change anything on the ValueLevel tab when de-duplicating.

However, we need to avoid removing any of the rows we need for building a composite where clause, so we must give them a grouped ID that is unique for the combined clause, such as [dataset].[variable].[seq]. Then we can de-duplicate.

De-duplicating Where Clauses in P21 v2

While you will most likely create your IDs manually, the following macro will auto-populate any empty ones with an ID made up of the dataset, variable, comparator and value. Any composite where clauses will be left alone if an ID is present, and then the sheet will be de-duplicated.

```
Sub WhereClauseIDs ()
```

```
'Select all rows without an ID
Sheet1.Range("A1:P10000").Select
ActiveCell.Offset(1, 0).Select
```

Filter WhereClauses tab to those without ID

```
'Insert formula into first empty cell
'<data>[Dataset].[Variable].[Comparator].[Value]</data>
Range("A1:P10000").ActiveCell.Offset(1, 0).Select
```

Find first empty cell

```
Do Until ActiveCell.EntireRow.Hidden = False
    ActiveCell.Offset(1, 0).Select
Loop
```

[Dataset].[Variable].
[Comparator].[Value]

```
'Fill down whole column
```

```
Range("B2").Select
Selection.FillDown
```

Fill down

```
'Paste as values
Range("A:P").AutoFilter
Range("A:P").Select
Application.CutCopyMode = False
```

Remove filter and paste as values

```
'De-duplicate sheet
ActiveCell.Offset(1, 0).Select
Header:=1, Columns:=5, Order:=1, Match:=False, Ascending:=True, Duplicates:=xlDuplicatesRemove
```

De-duplicate sheet

```
End Sub
```

Composite Where Clauses in P21 v3

In version 3, the ItemOID again consists of the clauses concatenated together (still without the ValueLevel variable name), but the WhereClauseOID uses the ID we define in the Where Clause tab.

This means we don't need to worry about duplicates appearing if we ensure our WhereClause ID is unique.

ID	Dataset	Variable	Comparator	Value
ADSK.DTYPE.2	ADSK	PARAMCD	EQ	K
ADSK.DTYPE.2	ADSK	AVISITN	IN	4,7,9,10,11,12,13,14,15
ADSK.DTYPE.2	ADSK	ATPTN	EQ	PRE-DIALYSIS


```
<ItemRef ItemOID="IT.ADSK.ADSK.PARAMCD.EQ.ADSK.AVISITN.IN.ADSK.ATPTN.EQ.412ea804ba920215685288c1808435ea8abd8892"
  OrderNumber="2"
  Mandatory="No"
  MethodOID="MT.ADSK.DTYPE.ADJISTAT">
  <def:WhereClauseRef WhereClauseOID="WC.ADSK.DTYPE.2" />
</ItemRef>
```

```
PARAMCD =
"K" (Potassium (mmol/L))
and
AVISITN IN (
4,
7,
9,
10,
11,
12,
13,
14,
15
) and
ATPTN = PRE-DIALYSIS
```

No Duplicate Where Clauses in P21 v3

In version 3, this is not an issue as the WhereClauseOID in the define.xml comes directly from the user-defined ID on the WhereClause tab.

Dataset	Variable	Where Clause
ADSK	STUDYID	ADSK.STUDYID.1
ADSK	STUDYID	ADSK.STUDYID.2
ADSK	USUBJID	ADSK.USUBJID.1
ADSK	USUBJID	ADSK.USUBJID.2

dy | Datasets | Variables | ValueLevel | WhereClause

ID	Dataset	Variable	Comparator	Value
ADSK.STUDYID.1	ADSK	PARAMCD	EQ	DIALKC
ADSK.STUDYID.2	ADSK	PARAMCD	NE	DIALKC
ADSK.USUBJID.1	ADSK	PARAMCD	EQ	DIALKC
ADSK.USUBJID.2	ADSK	PARAMCD	NE	DIALKC

Study | Datasets | Variables | ValueLevel | WhereClauses | Cc ...

```
<def:ValueListDef OID="VL.ADSK.STUDYID">
  <ItemRef ItemOID="IT.ADSK.ADSK.PARAMCD.EQ.dalc90e932dae0f306fb34091235d8611d9fb2ef"
    OrderNumber="1"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.STUDYID.1"/>
  </ItemRef>
  <ItemRef ItemOID="IT.ADSK.ADSK.PARAMCD.NE.dalc90e932dae0f306fb34091235d8611d9fb2ef"
    OrderNumber="2"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.STUDYID.2"/>
  </ItemRef>
</def:ValueListDef>
<def:ValueListDef OID="VL.ADSK.USUBJID">
  <ItemRef ItemOID="IT.ADSK.ADSK.PARAMCD.EQ.dalc90e932dae0f306fb34091235d8611d9fb2ef"
    OrderNumber="1"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.USUBJID.1"/>
  </ItemRef>
  <ItemRef ItemOID="IT.ADSK.ADSK.PARAMCD.NE.dalc90e932dae0f306fb34091235d8611d9fb2ef"
    OrderNumber="2"
    Mandatory="No">
    <def:WhereClauseRef WhereClauseOID="WC.ADSK.USUBJID.2"/>
  </ItemRef>
</def:ValueListDef>
```

```
<def:WhereClauseDef OID="WC.ADSK.STUDYID.1">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="EQ">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.ADSK.STUDYID.2">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="NE">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.ADSK.USUBJID.1">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="EQ">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
<def:WhereClauseDef OID="WC.ADSK.USUBJID.2">
  <RangeCheck SoftHard="Soft" def:ItemOID="IT.ADSK.PARAMCD" Comparator="NE">
    <CheckValue>DIALKC</CheckValue>
  </RangeCheck>
</def:WhereClauseDef>
```

No Duplicate Where Clauses in P21 v3

This means we don't need to de-duplicate, and from version 3.0.2 onwards, we no longer have the issue of multiple derivations appearing in the final column.

Variable	Where Condition	Label / Description	Type	Length or Display Format	Controlled Terms or ISO Format	Origin / Source / Method / Comment
STUDYID VLM		Study Identifier	text	11		See value-level metadata for variable definitions.
	PARAMCD = "DIALKC" (Dialysate Potassium Concentration (mmol/L))		text	11		Predecessor: FA.STUDYID
	PARAMCD ≠ "DIALKC" (Dialysate Potassium Concentration (mmol/L))		text	11		Predecessor: LB.STUDYID
USUBJID VLM		Unique Subject Identifier	text	20		See value-level metadata for variable definitions.
	PARAMCD = "DIALKC" (Dialysate Potassium Concentration (mmol/L))		text	30		Predecessor: FA.USUBJID
	PARAMCD ≠ "DIALKC" (Dialysate Potassium Concentration (mmol/L))		text	30		Predecessor: LB.USUBJID

In summary

Version 3.0.0 and 3.0.1 had an insurmountable issue that was fixed for 3.0.2. If you can't use v3.0.2, I recommend using version 2.2 with the stylesheet from version 3.

Version 2.2 needs a comment in order to display predecessors for ValueLevel items, and for WhereClauses to be de-duplicated but version 3 does not.

All versions require the Mandatory variables to be corrected.

The macros above can perform in seconds the fixes that previously took hours and allow for the simpler creation of a compliant define.xml

Any Questions?