PharmaSUG 2019 - Paper AD-107

Auto-generation of Clinical Laboratory Unit Conversions

Alan Meier, Cytel, Inc.

ABSTRACT

When mapping local labs to SDTM data structures, dealing with the free-text used by sites to indicate the original reporting units, in order to generate conversion factors to sponsor's standard reporting units is often a major headache.

Most lab units are in an "amount per volume" structure, where amounts are normally in some multiplier of mass (grams/moles), counts(cells), International Units or Enzyme Units and volumes are mainly in liters. This paper will look at a strategy to automate with SAS® much of the task through a three-step process: step 1 – parse and categorize the amounts and volumes, step 2 - normalize to a published conversion factor, and step 3 - calculate the final conversion factor.

INTRODUCTION

I've seen a number of approaches to dealing with lab unit conversions over the years, mainly involving one or more spreadsheets that have lab test names/codes, original raw data units, sponsor identified standard units and a conversion factor. These sheets can be at a study, project or even sponsor level and over time can get quite large as entries are needed for each variation of a unit (e.g. µg/L, ug/L, microg/L, mcg/L, Âµg/L, etc.). Maintaining them is often laborious and error prone.

What if the first time "mcg" is seen as an amount and is normalized to the standard "ug", that translation can be used for any analyte or volume. Further, it would be known that the base unit is "grams" and the prefix is "micro" with a multiplier of 10^{-6} (grams * 0.000001 = micrograms). This information would be extremely helpful in automating conversions.

This paper begins with a basic review of the "amount per volume" lab unit structure and basic conversion factor considerations from my 2014 PharmaSUG paper (IB-01 Challenges in Processing Clinical Lab Data). It then discusses the parsing of units on the incoming data and sponsor identified standards, along the way, examining a series of spreadsheets needed to identify and parse the units into base and prefix values for both amounts and volumes. The final step takes these values and automatically calculates sponsor standard results and normal ranges.

LAB UNIT BASICS

Regardless of whether the lab results are coming in from a central lab or from local labs, all labs need to supply units with the lab results. It's then the sponsor's responsibility to take those records and create SDTM original and standard results values in the LB domain. Even if the sponsor receives lab data in the standard units, they may be required to submit lab results in both U.S. conventional and Standard International (SI) units to different regulatory bodies. For these reasons, conversions are sometimes necessary.

Lab units come in a number of formats (ratios, excretion rates over time, etc.), but most are supplied in an "amount per volume" format (e.g. milligrams/liters). Conversions may be needed at either or both levels. The confusing part is knowing which changes increase the reported value and which decrease it.

For amounts, it's helpful to think of it as different sized packets and what is reported is the count of packets. The smaller the packet size, the higher the count and the larger the size, the smaller the count. If the original unit's amount is milligrams (smaller package/higher count) and the standard unit's amount is grams (larger package/smaller count), the conversion factor is going to be less than 1, in fact, 0.001. Going the opposite direction, grams to milligrams, the conversion factor is more than 1, in fact, 1,000.

Figure 1 - Amount Conversions is a visual representation of this.

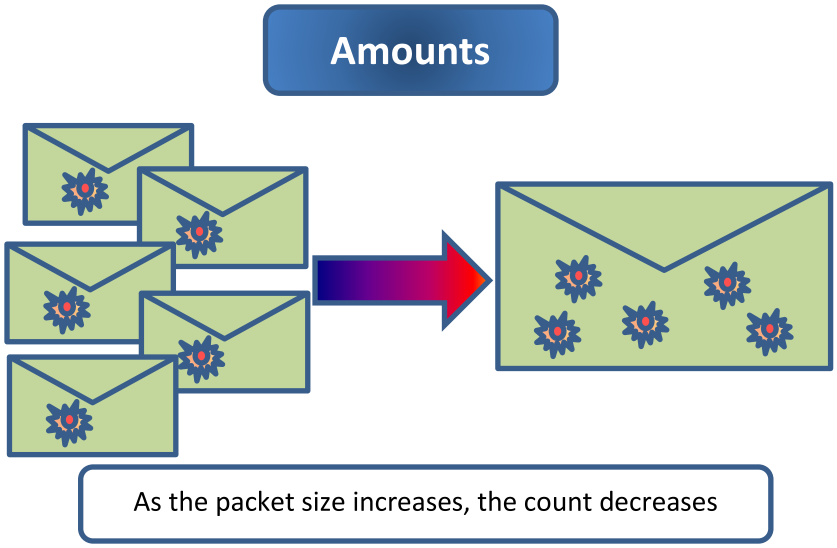

Figure 1 - Amount Conversions

Volumes are the opposite of amounts. Thinking of a jar, a gallon holds more volume than a quart and will have more of the analyte being measured. If the original unit's volume is milliliters (smaller) and the standard unit's volume is liters (larger), the conversion factor is going to be more than 1, in fact, 1,000. Going the opposite direction, liters to milliliters, the conversion factor is less than 1, in fact, 0.001.

Figure 2 - Volume Conversions is a visual representation of this.

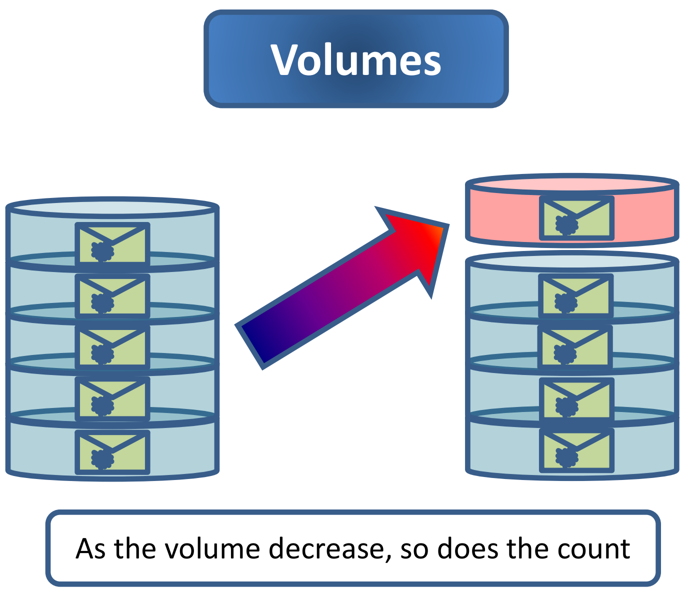

Figure 2 - Volume Conversions

PARSING UNITS

Drilling down further, amounts and volumes are usually made up of two parts, a base and a prefix. The prefix indicates a multiplier imposed on the base. Taking milligrams (mg) as an example, the base is "grams" and the prefix is "milli". With the prefix of "milli" defined, the multiplier of 0.001 is also known as a milligram is 1/1000 of a gram (or gram * 0.001). The parsing of amounts and volumes requires the first spreadsheet (***Unit_Standardization***) which decodes units into bases and prefixes. See Table 1 below.

Table 1 - Unit_Standardization

TYPE	ORIGINAL	STANDARD	BASE	PREFIX	ABBR	TERM	CAT
Amount/Volume	MG	MG	Grams	m	g	Milligrams	Amount
Amount/Volume	MILLIGM	MG	Grams	m	g	Milligrams	Amount
Amount/Volume	UG	UG	Grams	u	g	Micrograms	Amount
Amount/Volume	MCG	UG	Grams	u	g	Micrograms	Amount
Amount/Volume	µG	UG	Grams	u	g	Micrograms	Amount
Amount/Volume	NG	NG	Grams	n	g	Nanograms	Amount
Amount/Volume	ML	ML	Liter	m	L	Milliliter	Volume
Amount/Volume	UL	UL	Liter	u	L	Microliter	Volume
Amount/Volume	CUMM	UL	Liter	u	L	Microliter	Volume

After separating the incoming unit into amount and volume values, the table values are used. In the above table of ***TYPE*** 'Amount/Volumes', the upper-cased ***ORIGINAL*** units are entered. The ***STANDARD*** column provides a sponsor-defined normalized representation of the original values, identifying, for example, 'U', 'µ' and "MC' as 'micro' and translating it to a consistent ***PREFIX*** of 'u'. Also identified is the ***BASE*** as 'Grams' with an abbreviation (***ABBR***) and the ***TERM*** for the unit.

This is the primary spreadsheet where the majority of set up and maintenance is required. The savings on previous strategies, however, is enormous considering that there are already 15 permutations of the 5 amounts and 3 volumes displayed. Multiply that by the number of analytes that use grams/liter units (e.g. SODIUM, K, CHOL, HDL, LDL, GLUC, just to name a few) the number quickly grows. If a new amount of millimoles is needed, it's only one row to be added to the table and already 3 unique units are known and able to be converted.

The second spreadsheet (***Metric_Values***) provides the multipliers (see Table 2 below). Once it's set up, it won't need to be modified again.

Table 2 - Metric_Values

NAME	PREFIX	VALUE
zero		1.000E+00
deci	d	1.000E-01
centi	c	1.000E-02
milli	m	1.000E-03
micro	u	1.000E-06
nano	n	1.000E-09

Merging this spreadsheet with the previous data by ***PREFIX*** values, the multipliers are provided in ***VALUE***.

At this point, converting within a base value is relatively straight forward as it is only a matter of moving the decimal point. One approach is to determine a conversion factor to convert the values to the zero-prefix (base unit) and then again to convert that value to the new unit.

To demonstrate this, in order to convert a value in milligrams to micrograms, the value is multiplied by the multiplier 1.000E-03 and then divided by the multiplier 1.000E-06, based on the prefixes 'm' and 'u' respectively. Remember from the discussion in the previous section that when going from a smaller packet size for an amount to a larger packet size, the value gets smaller (milligrams to grams : value * 0.001) and reversed when going from a larger packet size to a smaller (grams to micrograms : value / 0.000001).

DEFINING STANDARD UNITS AND PUBLISHED CONVERSION FACTORS

The next spreadsheet needed, defines the sponsor's standard reporting units for each lab test, specimen type and analysis method (***Lab_Test***). A small segment is presented below in Table 3.

Table 3 - Lab_Test

LBTESTCD	LBTEST	LBSPEC	LBMETHOD	LBSTRESU	TYPE
GLUC	Glucose	SERUM		mmol/L	Amount/Volume
INSULIN	Insulin	SERUM		pmol/L	Amount/Volume
URATE	Urate	SERUM		mmol/L	Amount/Volume
WBC	Leukocytes	BLOOD		X10E9/L	Amount/Volume
WBC	Leukocytes	URINE	MICROSCOPIC	HPF	Microscopic
WBC	Leukocytes	URINE	AUTO-ANALYZER	X10E3/uL	Amount/Volume

The columns are straight out of the SDTM LB domain except for ***TYPE***. With ***LBSTRESU***, the starting and ending points are known. The last piece of the puzzle is getting from the original units to the standard units. That information comes in the final spreadsheet (***Conv_to_Std_CF***) as seen in Table 4.

Table 4 - Conv_to_Std_CF

	LBTESTCD	LBSPEC	LBMETHOD	LBSTRESU	TYPE	OUT_TYPE	LBSTCNVU	IN_TYPE
1	GLUC	SERUM		mmol/L	Amount/ Volume	Moles/Liter	mmol/L	Moles/Liter
2	GLUC	SERUM		mmol/L	Amount/ Volume	Moles/Liter	mg/dL	Grams/Liter
3	INSULIN	SERUM		pmol/L	Amount/ Volume	Moles/Liter	pmol/L	Moles/Liter
4	INSULIN	SERUM		pmol/L	Amount/ Volume	Moles/Liter	mIU/L	International Unit/Liter
5	INSULIN	SERUM		pmol/L	Amount/ Volume	Moles/Liter	ng/L	Grams/Liter
6	URATE	SERUM		mmol/L	Amount/ Volume	Moles/Liter	mg/dL	Grams/Liter
7	WBC	BLOOD		X10E9/L	Amount/ Volume	Count/Volume	X10E9/L	Count/Liter
8	WBC	URINE	AUTO-ANALYZER	X10E3/uL	Amount/ Volume	Count/Volume	X10E3/uL	Count/Liter
	LBTESTCD	LBSPEC	LBMETHOD	LBSTRESU	CF_REFU	CONV_FACTOR	CONV_ADD	SOURCE
1	GLUC	SERUM		mmol/L	mmol/L	1		
2	GLUC	SERUM		mmol/L	mmol/L	0.0555		Young & Huth
3	INSULIN	SERUM		pmol/L	pmol/L	1		
4	INSULIN	SERUM		pmol/L	pmol/L	6.945		Young & Huth
5	INSULIN	SERUM		pmol/L	pmol/L	0.172		Young & Huth
6	URATE	SERUM		mmol/L	umol/L	59.485		Young & Huth
7	WBC	BLOOD		X10E9/L	X10E9/L	1		
8	WBC	URINE	AUTO-ANALYZER	X10E3/uL	X10E3/uL	1		

To begin this table, each unique ***LBTESTCD***, ***LBSPEC***, and ***LBMETHOD*** should have a ***LBSTRESU*** entry with a published conversion factor input unit (***LBSTCNVU***) set to ***LBSTRESU*** and a conversion factor (***CONV_FACTOR***) set to 1. This allows the program to make the simple decimal place changes when the base amounts and volumes (***OUT_TYPE*** and ***IN_TYPE***) are the same. Usually, values other than 1 in the ***CONV_FACTOR*** and ***CONV_ADD*** columns are for when the base types don't match.

This paper doesn't go into how to identify published conversion factors when the base types don't match. I again refer you to my 2014 PharmaSUG paper for help on that. The identified published conversion factors are entered into ***CONV_FACTOR*** and when an algorithm requires an additional amount be added/subtracted after the published conversion factor is applied, that amount is entered in ***CONV_ADD***. As a documentation aid, the last column, ***SOURCE***, allows for a book or WEB-page reference.

DOING THE CONVERSIONS

All of the puzzle pieces are now available. The first step in calculating the final conversion factor is to parse all the lab units, including the raw data (***LBORRESU***), the sponsor identified standard unit (***LBSTRESU***), the published conversion factor input unit (***LBSTCNVU***) and published conversion factor output unit (***CF_REFU***).

Step two begins with matching base units from ***LBORRESU*** with the base units from ***LBSTCNVU***. Once the base units match, the raw values are converted to ***LBSTCNVU*** units as described above using the multipliers for both amounts and volumes.

Step 3 completes the process by applying the ***CONV_FACTOR*** and ***CONV_ADD*** values, followed by any additional conversions to go from published conversion factor output units (***CF_REFU***) to the sponsor identified standard reporting unit (***LBSTRESU***).

A walk-through example may help clarify the logic. The table below shows a URATE lab test for one subject where the original units (***LBORRESU***) are in milligm/L

USUBJID	LBTESTCD	LBORRES	LBORRESU	LBSPEC	LBMETHOD	VISIT	LBDTC
P0002	URATE	28.9	milligm/L	SERUM		BASELINE	2019-01-01

From the ***Lab_Test*** spreadsheet, based on matching ***LBTESTCD***, ***LBSPEC*** and ***LBMETHOD***, indicates that the standard reporting unit (***LBSTRESU***) is mmol/L and the type of unit (***TYPE***) is Amount per Volume.

LBTESTCD	LBTEST	LBSPEC	LBMETHOD	LBSTRESU	TYPE
URATE	Urate	SERUM		mmol/L	Amount/Volume

The information from the ***Unit_Standardization*** spreadsheet from below is read in and converted to format values based on the type Amount versus Volume as identified in the column ***CAT***.

TYPE	ORIGINAL	STANDARD	BASE	PREFIX	ABBR	TERM	CAT
Amount/Volume	G	G	Grams		g	Grams	Amount
Amount/Volume	MG	MG	Grams	m	g	Milligrams	Amount
Amount/Volume	MILLIGM	MG	Grams	m	g	Milligrams	Amount
Amount/Volume	L	L	Liter		L	Liter	Volume
Amount/Volume	DL	DL	Liter	d	L	Deciliter	Volume
Amount/Volume	MOL	MOL	Moles		mol	Moles	Amount
Amount/Volume	MMOL	MMOL	Moles	m	mol	Millimoles	Amount
Amount/Volume	UMOL	UMOL	Moles	u	mol	Micromoles	Amount

An upper-case version of the unit is created and the amount portion of the unit is separated from the volume (based on the location of the slash '/'). The upper-cased original amount "MILLIGM" is normalized to the sponsor standard of "MG" and further parsed to a base of "Grams" with a prefix of "m". The volume of "L" doesn't need to be normalized and is parsed to a base of "Liters" without a prefix. Based on the parsing, an ***IN_TYPE*** value of "Grams/Liter" is assigned.

Below is some sample code taken from a macro that does the above described parsing:

```
**** Parse Units into Amount and Volume ****;
&unit = upcase(&unit);
_i1_  = index (&unit, '/');
amt   = strip(substr(&unit, 1, (_i1_ - 1)));
vol   = strip(substr(&unit,    (_i1_ + 1)));
**** Parse Amount and Volume using Format Values****;
&prfx._amt_std = put(amt, $amt_std.);
&prfx._amt_bas = put(amt, $amt_bas.);
&prfx._amt_pre = put(amt, $amt_pre.);
&prfx._amt_abr = put(amt, $amt_abr.);
&prfx._amt_trm = put(amt, $amt_trm.);
&prfx._vol_std = put(vol, $vol_std.);
&prfx._vol_bas = put(vol, $vol_bas.);
&prfx._vol_pre = put(vol, $vol_pre.);
&prfx._vol_abr = put(vol, $vol_abr.);
&prfx._vol_trm = put(vol, $vol_trm.);
**** Create Type from Amount and Volume Terms ****;
in_type = strip(&prfx._amt_trm) || '/' || strip(&prfx._vol_trm);
```

This is repeated for the ***LBSTRESU*** value, resulting in an amount base of “Moles” and amount prefix of “m”. Again, the volume of “L” is parsed to a base of “Liters” without a prefix. Based on the parsing, an ***OUT_TYPE*** value of “Moles/Liter” is assigned.

From the ***Conv_to_Std_CF*** spreadsheet seen below, based on matching ***LBTESTCD***, ***LBSPEC***, ***LBMETHOD***, and ***IN_TYPE***, there is a “Grams/Liter” type published conversion factor (***CONV_FACTOR***) available. The published conversion factor converts input units in “mg/dL” (***LBSTCNVU***) to output units of “umol/L” (***CF_REFU***).

LBTESTCD	LBSPEC	LBMETHOD	TYPE	LBSTRESU	IN_TYPE	LBSTCNVU
URATE	SERUM		Amount/Volume	mmol/L	Grams/Liter	mg/dL

CONV_FACTOR	CONV_ADD	CF_REFU	OUT_TYPE	SORUCE
59.485		umol/L	Moles/Liter	Young and Huth

Once ***LBSTCNVU*** and ***CF_REFU*** are both parsed as described above, creation of a final conversion factor is now ready to begin. Here are all the steps, along with the URATE example values:

1. Convert original amount (***LBORRESU***) to base amount
   - Convert “mg” to “Grams” – Table 2 indicated the milligrams are 0.001 (1.000E-03) of a gram. This value was determined by using a format of prefixes and exponents on the amount prefix. With amounts, going from a smaller package size to a larger, the value gets smaller, so, the first conversion factor (***CF1***) is 1 * 0.001.
2. Convert base amount to the published conversion factor input amount (***LBSTCNVU***)
   - Convert “Grams” to “mg” – Here the inverse is done to return to the original amount, so, the second conversion factor (***CF2***) is 1 / 0.001 or 1000.
3. Convert original volume (***LBORRESU***) to base volume
   - Convert “L” to “Liters” – L is already the base unit, so the third conversion factor (***CF3***) is 1.
4. Convert base volume to the published conversion factor input volume (***LBSTCNVU***)
   - Convert “Liters” to “dL” – Table 2 indicated the deciliters are 0.1 (1.000E-01) of a liter. With volumes, going from a larger size to a smaller, the value gets smaller, so, the fourth conversion factor (***CF4***) is 1 * 0.1.
5. Convert published conversion factor output amount (***CF_REFU***) to the base amount

- Convert “umol” to “Moles” - Table 2 indicated the micromoles are 0.000001 (1.000E-06) of a mole. With amounts, going from a smaller package size to a larger, the value gets smaller, so, the fifth conversion factor (***CF5***) is 1 * 0.000001.

6. Convert the base amount to the sponsor identified standard amount (***LBSTRESU***)
   - Convert “Moles” to “mmol” - Table 2 indicated the millimoles are 0.001 (1.000E-03) of a mole. With amounts, going from a larger package size to a smaller, the value gets larger, so, the sixth conversion factor (***CF6***) is 1 / 0.001 or 1000.
7. Convert published conversion factor output volume (***CF_REFU***) to the base volume
   - Convert “L” to “Liters” – L is already the base unit, so the seventh conversion factor (***CF7***) is 1.
8. Convert the base volume to the sponsor identified standard volume (***LBSTRESU***)
   - Convert “Liters” to “L” – L is already the base unit, so the eighth conversion factor (***CF8***) is 1.
9. Multiple the results from the above 8 steps with the published conversion factor (***CONV_FACTOR***)
   - ***CF1*** * ***CF2*** * ***CF3*** * ***CF4*** * ***CF5*** * ***CF6*** * ***CF7*** * ***CF8*** * ***CONV_FACTOR***)
   - 0.001 * 1000 * 1 * 0.1 * 0.000001 * 1000 * 1 * 1 * 59.485 = 0.0059485

This final conversion factor can now be used for URATE result values and the normal ranges to populate the SDTM ***LBSTRESC***, ***LBSTRESN***, ***LBSTNRHI*** and ***LBSTNRLO*** fields in ***LB*** where the incoming original units are “milligr/L”.

CONCLUSION

The focus of this paper was on Amount/Volume unit conversions. The premise can be extended to cover excretion rates by adding a time portion (Amount/Unit/Time). Percent/Ratio conversion are also easy to add.

A more complex algorithm could be devised to skip the middle steps of going to base units, but I prefer keeping the conversion simple.

To reduce run times, unique ***LBTESTCD***, ***LBORRESU***, ***LBSPEC***, and ***LBMETHOD*** combination values should be identified before the process to identify conversion factors is initiation. Conversion factors can then be merged back onto the full lab data set by the same 4 variables.

If a sponsor needs to maintain multiple standard reporting units (e.g. U.S. Conventional Units and S.I. Units) in order to provide data to multiple regulatory agencies, a second ***Lab_Tests*** spreadsheet is needed. Additional conversion factors can be added to ***Conv_to_Std_CF***, but merges will need to include ***LBSTRESU*** in order to identify the sponsor identified output units.

APPENDIX

Below is a table (Table 5) of the spreadsheets used, with the column labels and a description.

Table 5 - Appendix Table of Spreadsheets

Spreadsheet	Column	Description
Unit_Standardization	***TYPE***	A classification of where the unit is used (e.g. Amount/Volume, Ratio, etc.)
	ORIGINAL	Upper-cased version of Raw unit
	STANDARD	Raw unit, normalized to sponsor's preferred terminology
	BASE	Underlying base of the unit (e.g. grams, liters, etc.)
	PREFIX	The proper case prefix of the unit used to determine the multiplier on the base unit value (e.g. m=milli-, u=micro-, etc.)
	ABBR	Proper case terminology of the base value
	TERM	The English written version of the unit
	CAT	A categorization of the unit as to its use (e.g. Amount, Volume, International Unit, etc.)
	NOTES	A field to record any notes deemed helpful in review
Metric_Values	***NAME***	The English written version of the prefix
	PREFIX	The proper case metric prefix
	VALUE	The exponent multiplier of the base
Lab_Tests	***LBTESTCD***	SDTM Lab Test Code
	LBTEST	SDTM Lab Test Name
	LBSPEC	SDTM Lab Test Specimen Type
	LBMETHOD	SDTM Lab Test Methodology
	LBSTRESU	SDTM Lab Test Standard Result Unit
	TYPE	A classification of where the unit is used (e.g. Amount/Volume, Ratio, etc.)
Conv_to_Std_CF	***LBTESTCD***	SDTM Lab Test Code
	LBSPEC	SDTM Lab Test Specimen
	LBMETHOD	SDTM Lab Test Methodology
	TYPE	A classification of where the unit is used (e.g. Amount/Volume, Ratio,, etc.)
	LBSTRESU	SDTM Lab Test Standard Result Unit
	IN_TYPE	The base unit type definition of the input unit to a Published Conversion Factor (e.g. Grams/Liter)
	LBSTCNVU	The input unit to a Published Conversion Factor (e.g. mg/L)
	CONV_FACTOR	A Published Conversion Factor multiplier for a given input unit (LBSTCNVU) and output unit (CF_REFU)
	CONV_ADD	An amount added to a value after a Published Conversion Factor multiplier has been applied to a value
	CF_REFU	The output unit to a Published Conversion Factor (e.g. mmol/L)

Conv_to_Std_CF (cont.)	***OUT_TYPE***	The base unit type definition of the output unit to a Published Conversion Factor (e.g. Moles/Liter)
	SOURCE	A field to document the source of any Published Conversion Factor

REFERENCES

Meier, Alan. 4-Jun-2014. "Challenges in Processing Clinical Lab Data." *Proceedings of the PharmaSUG 2014 Conference*, San Diego, CA : PharmaSUG. Available at https://pharmasug.org/proceedings/2014/IB/PharmaSUG-2014-IB01.pdf

Young, Donald S. and Edward J. Huth. 1998. "SI Units for Clinical Measurements." Philadelphia, PA : American College of Physicians

ACKNOWLEDGMENTS

I'd like to acknowledge and thank those individuals who reviewed this paper and provided feedback.

RECOMMENDED READING

- The International System of Units – NIST Special Publication 330 (https://www.nist.gov/pml/special-publication-330)

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Alan Meier
Cytel, Inc.
Alan.Meier@Cytel.com