Paper TT05

The automated unit verification, units conversion, and converted results verification through the Unified Code for Units of Measure and LOINC

Sam Tomioka, Sunovion Pharmaceuticals, Fort Lee, NJ USA

ABSTRACT

The verification of scientific units and conversion from the reported units to standard units have always been challenging during the preparation of CDISC SDTM LB and/or ADaM ADLB due to several reasons especially when the study involves multiple local clinical laboratories. Despite the challenges, the lookup table approach is the norm for many companies for verification of the units, unit conversions and verification of results in the new units. In this paper, I propose an alternate method for units verification, units conversion, and verification of converted results leveraging the Unified Code for Units of Measure based API, and provide evaluation results of two UCUM based API services, and use cases in the clinical laboratory data stored in SDTM.LB.

INTRODUCTION

The verification of scientific units and conversion from the reported units to standard units have always been challenging during the preparation of CDISC SDTM LB and/or ADaM ADLB due to several reasons especially when the study involves multiple local clinical laboratories. Despite the challenges, the approach that uses lookup table with test names, test codes, and conversion factors is the norm for many companies for verification of the units, unit conversions and verifications of results in the new units [1, 2, 3, 4]. This approach requires considerable amount of maintenance overhead as it needs to include all possible unit expressions along with the conversion factors. More recently, consideration was given for a more systematic approach that does not require to include all possible units in the lookup table but maintaining only the tables for unit normalization and metric conversion [5]. While this approach works on amount base unit conversion and reduces the maintenance effort, it does not eliminate the lookup table, and may not work well on molar mass base unit conversions. The regulatory agencies require the sponsors to use standardized units for reporting and analysis [6]. The PMDA requires SI units for all reporting and analysis [7,8]. The differences in regulatory requirements have forced us to maintain region specific conversion for some measurements which add additional complexity. One of the limitations with this approach is that unless the lookup table consists of all possible combinations of tests (`LBCAT`/`LBTESTCD`), reported units, the standard units, and the conversion factors, some records will not be converted. This approach is especially troublesome when `LBTESTCD`, `LBSPEC`, and `LBMETHOD` are mapped to more than one molar mass and do not distinctively identify the laboratory test across multiple studies.

In this paper, we propose an alternate method for unit verifications, unit conversions, and verification of converted results leveraging the Unified Code for Units of Measure (UCUM) based API. The unit conversions and results validation through UCUM API were originally proposed and tested by Jozef Aerts on FDA-CDISC pilot 2013 dataset using Java in early 2019 [9]. This approach eliminates the need for the lookup table.

The UCUM has been developed and maintained by Regenstrief Institute, Inc. and the UCUM Organization.[10] The UCUM has been implemented by many organizations such as FDA, LOINC, DICOM, IEEE, FHIR, and OHDSI is implementing UCUM for OMOP. It is a code system that includes all units of measures being used in science, engineering and business designed to facilitate machine readable results and their units.

There are two free UCUM verification and conversion RESTful API services available and they are slightly different. One is the production version of API services provided by The National Library of Medicine and the second API services are provided as the development version by XML4Pharma

[11,12]. The National Library of Medicine also provides services through web browser where users can manually validate and convert UCUM units, and batch validate UCUM units. XML4Pharma has recently published the improved version 2 of the API services which covers various type of unit conversions, therefore this paper mainly focuses on the evaluation of the API services from XML4Pharma. Integrated SDTM LB data were used for this evaluation.

METHOD

This section discusses how the RESTful API services were implemented using Python to facilitate the automation of unit verifications, unit conversions, and results verifications for the lab data leveraging LOINC codes. The link to the source scripts is provided.

Conversion to UCUM Notation

In order to use the API services, the units stored in the data must be converted to UCUM notation. One of the challenges of using UCUM based conversion comes from the SDTM LB itself as it does not allow UCUM as unit expression, instead it must use the CDISC SDTM UNIT codelist. CDISC has published the UNIT-UCUM mapping dictionary at 'Terminology' page. The mapping dictionary lists all published UNIT and the corresponding UCUM including some non-unit of measures such as 'BAG', 'CAN','DRINK', 'PACKAGE' [13]. While one could utilize the CDISC mapping file to update the target units represented in CDISC UNIT, it cannot be used for the reporting units which are often not conformant to CDISC UNIT. Therefore, the dictionary file from CDISC won't be the complete solution to address the challenge.

In this evaluation, regular expressions were used to convert the reporting units and the target units in UNIT to UCUM notation as shown in *Script 1* below. Each tuple element consists of characters from the source unit, and the characters in the target unit.

```
01 patterns = [("%","%25"),
02             ("\A[xX]?10[^E]", "10*"),
03             ("IU", "%5BIU%5D"),
04             ("\Anan", ""),
05             ("\ANONE", ""),
06             ("\A[rR][Aa][Tt][Ii][Oo]", ""),
07             ("\ApH", ""),
08             ("Eq[l]?","eq"),
09             ("\ATI/L","TR/L"),
10             ("\AGI/L","GA/L"),
11             ("V/V","L/L"),
12             ("[a-z]{0,4}/HPF","/%5BHPF%5D"),
13             ("[a-z]{0,4}/LPF","/%5BLPF%5D"),
14             ("fraction of 1","1"),
15              ("sec","s"),
16              ("1.73m2","%7B1.73_m2%7D"),
17              ("\AG/L","GA/L")
18             ]
19 _, ucumList = orresu2ucum(sdtmLb,patterns)
```

Script 1 Example of regular expression.

Passing the SDTM.LB and `patterns` of regular expression to `orresu2ucum` generates `ucumList`, which is the list of UCUM notation.

Some characters passed to the API call required to be URL encoded. For example, `[` must be expressed as `%5B`. In UCUM notation, annotations are enclosed by two brackets `{}`, and these are expressed as `%7B` and `%7D`. Above lines 12, 13, and 16 address this encoding. Table 1 shows a few examples of URL encoding.

Table 1 A few examples of URL encoding

!	#	$	%	&	'	(	)	*	+	,	/	:	;	=	?	@	[	]
%21	%23	%24	%25	%26	%27	%28	%29	%2A	%2B	%2C	%2F	%3A	%3B	%3D	%3F	%40	%5B	%5D

Validation of UCUM Notation

Next step after conversion of unit expression in SDTM LB to UCUM is the validation of UCUM notation. This can be easily completed using 'The UCUM unit validity service' provided by the US National Library of Medicine. Here is the example of how one could call the verification API for `mm[Hg]` using the web browser. It returns Boolean response.

```
https://ucum.nlm.nih.gov/ucum-service/v1/isValidUCUM/mm%5BHg%5D
```

Script 2 API call for verification of `mm[Hg]`

The above API call was wrapped in `ucumVerify` function.

```
01  url='http://xml4pharmaserver.com:8080/UCUMService2/rest'
02  ucumVerify(ucumList, url)
03
04
05  Out: ['g/dL = true',
06        'mg/dL = true',
07        'ng/mL = true',
08        '10*3/uL = true',
09        '%25 = true',
10        '10*6/uL = true',
11        '10*3/mm3 = true',
12        'meq/L = true',
13        'mL/min = true',
14        'ng/L = true',
15        'ng/dL = true',
16        'u%5BIU%5D/mL = true',
17                  ⋮
```

Script 3 Example of using a defined function ucumVerify to verify the list of UCUM notations.

Conversion from LBORRESU to LBSTRESU

Conversion services are provided by The National Library of Medicine and XML4Pharma.

The National Library of Medicine provides version 1 of API services which have some limitations. For example, it does not have concept of molar mass, it doesn't know the LOINC, and it does not account for valence of ion of the specific lab tests. XML4Pharma API services have been updated to accommodate these concepts recently. It verifies the validity of LOINC before the molar mass based conversion, and it accounts for valence of ion when the conversions involve 'Eq' to molar unit.

The `convert_unit` function takes the SDTM LB as the first parameter (Script 4). This data should include `LBTESTCD`, `LBTEST`, `LBORRES`, `LBORRESU`, `LBSTRESU`, `LBSTRESN`, and `LBLOINC`. Variables `LBTESTCD`, `LBTEST`, `LBORRES`, `LBORRESU`, `LBSTRESU`, and `LBLOINC` are passed to the API. `LBSTRESN` is compared against the API response (Figure 1). The second parameter is the `url` for API, this parameter `patterns` is the regular expression discussed earlier, and the fourth parameter `loinconly` accepts either [0 or 1] where 0 would convert all records in the input data, and 1 returns conversion for records with LOINC. This function takes the SDTM LB, converts `LBORRESU` and `LBSTRESU` in UCUM notation, performs conversion of `LBORRES` to `LBSTRESN`, and returns 3 objects. The first object contains the discrepant records between `LBSTRESN` and the results obtained from the API call, the second object contains all the records including the converted results,

and the third object contains the original API responses. Essentially, this is the only function needed for lab data conversion and validation, but it is recommended to run `ucumVerify` first to ensure that the regular expression can produce the accurate UCUM notation.
The original API response is provided in the semi-structured data (XML).


```
<UCUMWebServiceResponse ServerDateTime="2019-12-08T17:52:00">
  <WebServiceRequest>
    http://xml4pharmaserver.com:8080/UCUMService2/rest/ucumtransform/118/from/mg/dL/to/mmol/L/LOINC/18262-6
  </WebServiceRequest>
  <Response>
    <SourceQuantity>118.0</SourceQuantity>
    <SourceUnit>mg/dL</SourceUnit>
    <TargetUnit>mmol/L</TargetUnit>
    <MolecularWeightUsed>386.6535</MolecularWeightUsed>
    <ResultQuantity>3.0518281</ResultQuantity>
  </Response>
</UCUMWebServiceResponse>
```


Figure 1 API response

Example call of `convert_unit` and the output of findings data are provided below.

```
01    findings,full,response=convert_unit(nodupdf, url, patterns,loinconly=0)
```

Out:

```
findings[(findings['fromucum'].notnull())]
```

	LBTESTCD	LBTEST	LBORRES	LBORRESU	LBSTRESU	LBSTRESN	LBLOINC	checklist	fromuc
1	BILI	Bilirubin	0.4	mg/dL	umol/L	6.00	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	6.841
2	CREAT	Creatinine	0.96	mg/dL	umol/L	85.00	2160-0	http://xml4pharmaserver.com:8080/UCUMService2/...	84.867
4	GLUC	Glucose	92	mg/dL	mmol/L	5.10	2345-7	http://xml4pharmaserver.com:8080/UCUMService2/...	5.105
6	URATE	Urate	4.5	mg/dL	mmol/L	0.27	3084-1	http://xml4pharmaserver.com:8080/UCUMService2/...	0.267
7	UREAN	Urea Nitrogen	10	mg/dL	mmol/L	3.70	3094-0	http://xml4pharmaserver.com:8080/UCUMService2/...	3.571
21	BILI	Bilirubin	0.6	mg/dL	umol/L	10.00	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	10.262

Script 4 Example call of `convert_unit`

Findings data contains the records with inconsistent results between `LBSTRESN` and the last column which contains the values generated by the API services.

EVALUATION

Conversion results

Most of the converted results from the API responses were consistent with the conversions performed by the lab vendors. While there were some discrepancies identified, all of these were due to incorrect conversions performed by the lab vendors.

A total of 17565 unique lab records were tested for this review using the API services provided by XML4Pharma. All are unique records based on `LBTESTCD`, `LBORRES`, `LBORRESU`, and `LBSTRESU`. The following Figure 2 shows the maximum normalized differences between the source and the UCUM conversions when they are not equal. There are many tests with negligible differences.

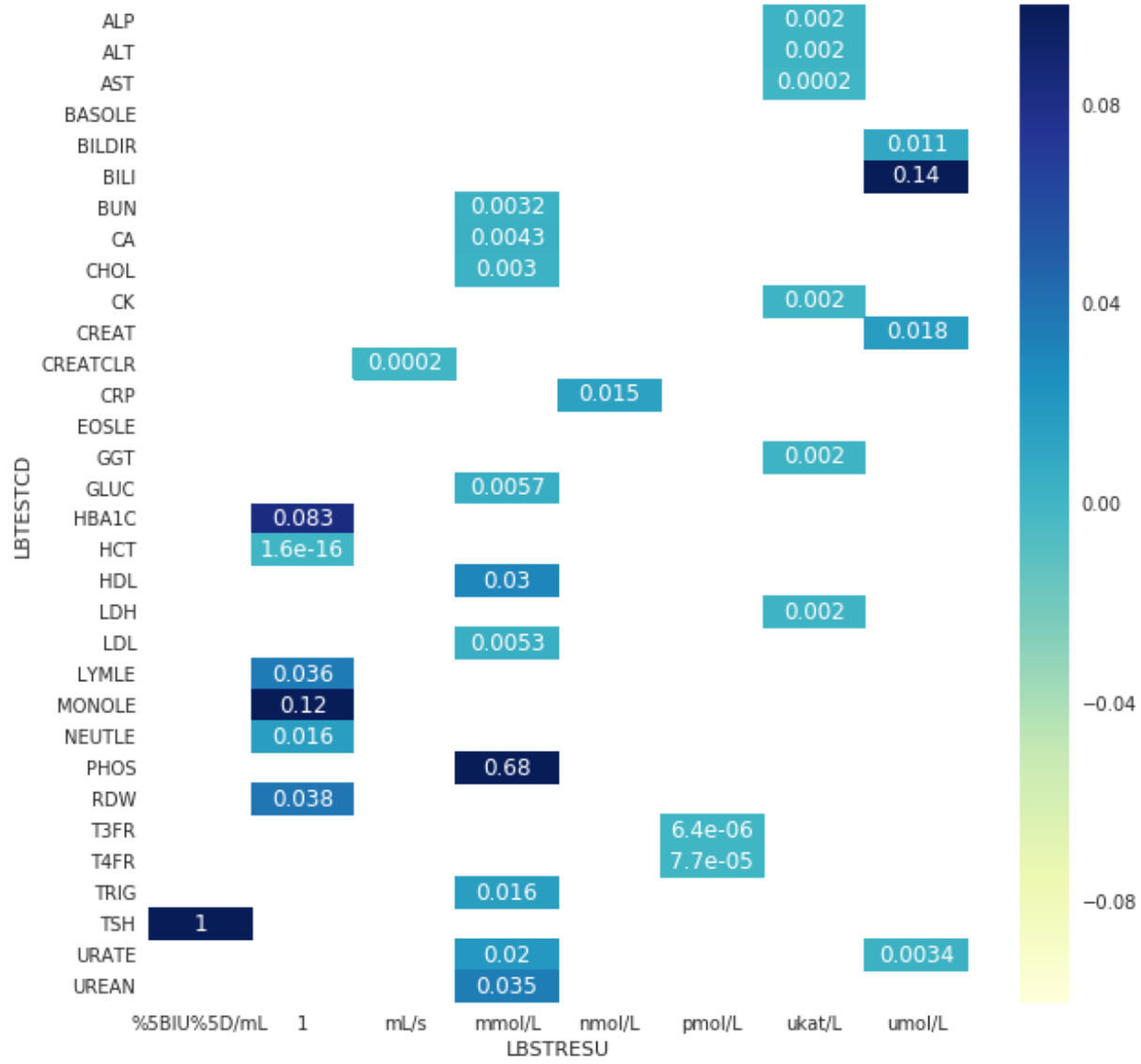

Figure 2 Normalized maximum differences between source and UCUM API responses

HBA1C has the mean normalized difference of 4.7% and the maximum difference of 8.3%. These differences were caused by the difference in the rounding. The differences found in MONOLE were due to the rounding differences.

Magnesium is one of the tests which requires consideration of the ion valence during the unit conversion. This evaluation confirmed that the API services can correctly convert such laboratory tests when LOINC is included in the API call.

The source data had known issues with TSH and PHOS, and it is clear from Figure 2 above that these have discrepancies between LBSTRESN and the API responses. PHOS is one of the tests often incorrectly converted because PHOS is used interchangeably for phosphate and phosphorus. These two analytes have different molar masses.

The differences in BILI were not related to the rounding difference. This is a newly identified conversion issue during this evaluation.

Table 2 First few records of BILI in `finding` data

LBTESTCD	LBTEST	LBORRES	LBORRESU	LBSTRESU	LBSTRESN	LBLOINC	checklist	fromucum
BILI	Bilirubin	0.4	mg/dL	umol/L	6.0	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	6.841559
BILI	Bilirubin	0.6	mg/dL	umol/L	10.0	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	10.262338
BILI	Bilirubin	0.2	mg/dL	umol/L	4.0	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	3.420779
BILI	Bilirubin	0.6	mg/dL	umol/L	11.0	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	10.262338
BILI	Bilirubin	0.4	mg/dL	umol/L	7.0	1975-2	http://xml4pharmaserver.com:8080/UCUMService2/...	6.841559

For example, 0.4mg/dL was reported as 6.0 µmol/L in `LBSTRESN` for the first record. The API returned it as 6.841559. The molar mass is 584.6621g/mol for BILI based on LOINC 1975-2, therefore

$$0.4\frac{mg}{dL} = 0.004\frac{g}{L} \cdot \frac{\mu mol}{584.6621 \cdot 10^{-6} g} = 6.8415585 \ldots \frac{\mu mol}{L}$$

The API services were able to correctly identify the incorrect conversion in the source data.

Error messages in the API responses

There were cases where the API conversions were unsuccessful.

The most frequent error response was `ERROR: invalid double for Molecular Weight value = null.` This is caused by either the missing molar mass or LOINC code in the source data when the conversion requires molar mass. This error can be addressed by providing LOINC for all tests performed. This can be challenging when the data is provided by small local laboratories. Alternatively, this error can be addressed by providing the molar mass of the analyte.

There were two kinds of `No MW found for LOINC Part Number LPxxxx for LOINC code yyyy.` Specially, `ERROR: No MW found for LOINC Part Number LP15449-9 for LOINC code 15153-0` and `ERROR: No MW found for LOINC Part Number LP15449-9 for LOINC code 35192-4` were returned during this evaluation. However, as of writing this paper, the API calls returned the converted results. It appears that the molar mass data in LOINC have been expanded in the past few months.

Creatinine Clearance (mL/min/1.73 m2) conversion failed with `ERROR: number of annotations in source and target is different`. The surface area 1.73 m^2 was added as annotation in the source but the target did not include the same annotation. This can be solved by adding the same annotation in the target unit. For example,

```
http://xml4pharmaserver.com:8080/UCUMService2/rest/ucumtransform/121/from/mL/min/%7B1.73_m2%7D/to/mL/s/%7B1.73_m2%7D
```

Script 5 Updated API call which includes annotation in source and target.

Alternatively, the annotation can be dropped from the API call because the API ignores the annotations.

Several conversions failed with `ERROR: unexpected result: Error: Source and Target unit do not seem to belong to the same property`. These are listed in Table 3. The UCUM API seems to have issue with `G` (giga) and `T `(Tera). The same error message was returned even after the second character was added (GA, TR). `G` or `Giga per liter` is often used in the hematology panel and is equivalent to 10*9 as per ucum-essence.xml, but the conversion was not successful [14]. `T` and `10*12` are equivalent as per ucum-essence.xml, but the conversion failed.

```
http://xml4pharmaserver.com:8080/UCUMService2/rest/ucumtransform/1/from/G/L/to/10*9/L
http://xml4pharmaserver.com:8080/UCUMService2/rest/ucumtransform/1/from/GA/L/to/10*9/L
```

Script 6 Reproducible API call to get the error message.

Furthermore, the ucum-essence.xml suggests that %5BIU%5D i.e. [IU] and U are equivalent, but the conversion was not successful.

Table 3 Unsuccessful Conversion

Index	LBTESTCD	LBORRESU	LBSTRESU
5652	INSULIN	uU/mL	pmol/L
6586	BASO	G/L	10*9/L
6588	EOS	G/L	10*9/L
6591	LYM	G/L	10*9/L
6594	MONO	G/L	10*9/L
6597	NEUT	G/L	10*9/L
6600	PLAT	G/L	10*9/L
6606	RBC	T/L	10*12/L
6609	WBC	G/L	10*9/L
7026	LYMAT	G/L	10*9/L
7828	MYCY	G/L	10*9/L
11395	ALP	%5BIU%5D/L	U/L
11400	ALT	%5BIU%5D/L	U/L
11403	AST	%5BIU%5D/L	U/L
11416	CK	%5BIU%5D/L	U/L

Summary

As of writing this paper, all the molar mass-based conversions were successful when LOINC is included in the API call. It is highly recommended to obtain LOINC codes for all laboratory tests performed in the clinical trials, but it will be one of the FDA's standard data requirements as early as March 15, 2020. The API services can be used to identify incorrect conversions provided by the vendors. There are some challenges with regards to the units containing [IU], U, G, and T. As of now, these can be addressed through building a small dictionary. For example, G -> 10^9 .The conversion which involves 'Eq' was successful.

This API based methods can replace the lookup table approach when the source data contains LOINC codes if not the molar mass of the analyte, and can be used for the purpose of conversion and validation.

All the scripts and functions discussed here are available at https://github.com/stomioka/ucum

REFERENCE

[1] Fulton, J., "Dealing with Lab Data – Stacking the Deck in Your Favor", *SAS Global Forum*,

182-2010, 2010. Available at: http://support.sas.com/resources/papers/proceedings10/182-2010.pdf
[2] C. Z. Wu and D. Wales, "Laboratory Data Standardization with SAS", *PharmaSUG*, AD10, 2017. Available at: https://www.pharmasug.org/proceedings/2017/AD/PharmaSUG-2017-AD10.pdf
[3] Filar, K and Kuczkowska, P., "The First Steps in Laboratory Dataset", *PhUSE*, PP09, 2017. Available at: https://www.phusewiki.org/docs/Conference%202017%20PP%20Papers/PP09.pdf
[4] Bryant, M. A., Pemble, P., Franchino, J., et.al., "Practices – Standardizing Laboratory Units within a Sponsor/Submission White Paper" *PhUSE*, 2018. Available at: http://www.phusewiki.org/wiki/images/5/50/Laboratory_Units_Best_Practices_FV_16Jan2015.docx
[5] Meier, A., "Auto-generation of Clinical Laboratory Unit Conversions", *PharmaSUG*, Paper AD107, 2019. Available at: https://www.lexjansen.com/pharmasug/2019/AD/PharmaSUG-2019-AD-107.pdf
[6] "Position on Use of SI Units for Lab Tests." *U.S. Food & Drug Administration*, 2013. [Online] Available: https://www.fda.gov/media/109533/download. [Accessed: 08- Dec- 2019].
[7] "Notification on Practical Operations of Electronic Study Data Submissions." Ministry of Health, Labour and Welfare, 2015.
[8] "FAQs on Electronic Study Data Submission (Excerpt)", 2019. [Online]. Available at: https://www.pmda.go.jp/english/review-services/reviews/0007.html. [Accessed: 08- Dec- 2019].
[9] Aerts, J. *SDTM --STRESN: why we need UCUM*. 2019. [online] Available at: http://cdiscguru.blogspot.com/2019/02/sdtm-stresn-why-we-need-ucum.html [Accessed 9 Apr. 2019].
[10] G. Schadow and C. McDonald, "The Unified Code for Units of Measure", *UCUM*, 2017. [Online]. Available at: http://unitsofmeasure.org/trac. [Accessed: 08- Apr- 2019].
[11] Unified Code for Units of Measure (UCUM) at NLM. [Online]. Available at: https://ucum.nlm.nih.gov/.
[12] Aerts, J. "Instructions for testing the RESTful web service for converting "molar" concentrations to "mass" concentrations and vice versa using UCUM units and LOINC codes." 2019 [Online]. Available at: http://www.xml4pharma.com/UCUM/Instructions_for_testing_the_RESTful_web_service_for_molar_mass_unit_conversions.pdf
[13] "UCUM and CDISC Codelists", *CDISC*, 2019. [Online]. Available at: https://www.cdisc.org/kb/article/ucum-and-cdisc-codelists. [Accessed: 09- Apr- 2019].
[14] "ucum-essence.xml". [Online]. Available at: http://unitsofmeasure.org/ucum-essence.xml.