Paper DS15

# SDTM UP – Versioning – Do's & Don'ts

Utkarsh Pathak, Ephicacy Lifescience Analytics, Bengaluru, India
Charumathy Sreeraman, Ephicacy Canada Inc., Toronto, Canada

## ABSTRACT

Regulatory submissions are required to comply with or endorse current versions of the CDISC standards, which are continuously updated. More standardized data must be incorporated into the new standards version, which will help with its reuse in different submissions and purposes. This article aims to provide insights to organizations who are putting new standards into practice. The authors of this paper share some challenges and best practices they encountered during the process of up-versioning to CDISC standards. We will also highlight about the utility-driven conversion of SDTM 3.2 datasets to higher CDISC standards (i.e. SDTM 3.3/SDTM 3.4). Provide knowledge on how to read and validate these SDTM outputs in accordance with the conformance standards.

## INTRODUCTION

SDTM up-versioning refers to the process of refreshing the version of the CDISC Study Data Tabulation Model (SDTM). This model outlines how clinical study data is formatted for submission to regulatory bodies. For recent studies sent to regulatory agencies, it is typically advisable to utilize the most current version of the SDTM. In the case of long-term studies, transitioning to the most recent version of the SDTM may be prudent. The benefits of up-versioning include ensuring high-quality data compliance and aligning studies with relevant timelines and publications.

This paper outlines the essential methods and procedures that must be adhered to in a semi-automated manner for transitioning from one SDS version to a higher version. The overall procedure is as follows:
Identify modifications: Analyze the differences between the existing and new versions of the SDTM.
Determine which versions to refresh: Update those versions that have undergone changes since the last published iteration of the SDTM.
Consider alignment: Verify that the SDTM is consistent with other standards.

## CDISC AND ITS EVOLUTION

CDISC standards have emerged as a crucial part of the clinical research process since they provide a framework for standardizing and organizing data across different clinical studies. The U.S. Food and Drug Administration (FDA) and other regulatory agencies demand that data be submitted in a consistent format that conforms to CDISC standards. This ensures that the information supplied is reliable, uniform, and simple for regulatory bodies to review and evaluate.

The increasing complexity of clinical research and technological improvements are driving the ongoing growth of CDISC standards. To handle new opportunities and difficulties in the area, CDISC is always trying to improve current standards and create new ones. This guarantees that CDISC standards continue to be applicable and useful in a constantly evolving field of research.

## NEED FOR UPVERSIONING

SDTM up-versioning is necessary to maintain to consistency and make the data standard applicable to evolving clinical trial practices. New data types, improved analysis methods, and enhanced data capture capabilities can all be added as a result. Regulatory agencies like the FDA mandate that studies be submitted using the most recent version of the SDTM standard in order to preserve consistency across submissions and expedite review processes.

## PROCESS FLOW

The flow diagram below describes the overall conversion process.

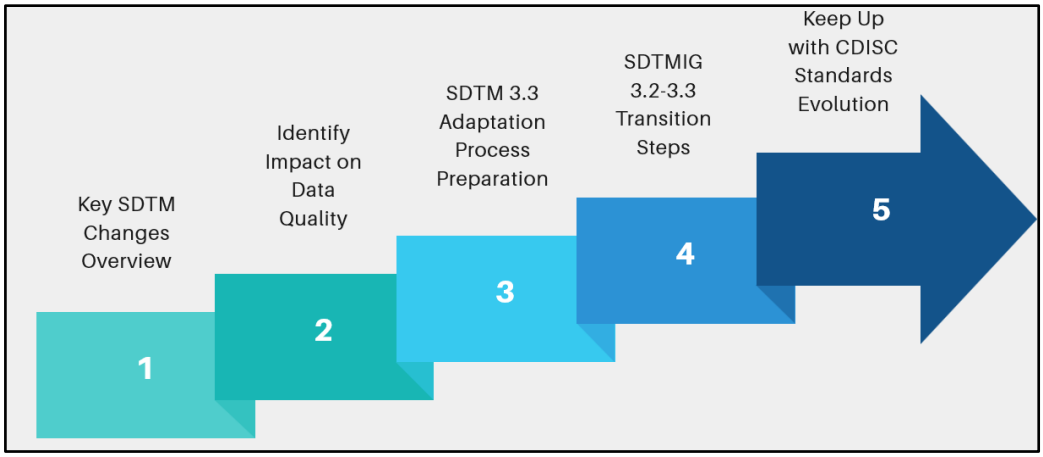


*Figure 1. Process flow for Up - Versioning*

The SDTM 3.2 datasets or xpts, which are produced by the sponsor or by an existing mapping process, serve as the input for the conversion process. The purpose of the gap report is to examine the modifications that should be made during the conversion process. To obtain the differences between the versions, metadata datasets are generated using the SDTM IG v3.2/v3.3 as a basis.

Any programming languages that are accessible or agreed upon must then be used to develop the utilities. After that, we validate the modified SDTM datasets both programmatically and manually. We further confirm that the data reflects proper metadata updates by the Enterprise/Community versions of Pinnacle21. In this paper, we analyzed the data using SAS/STAT® software.

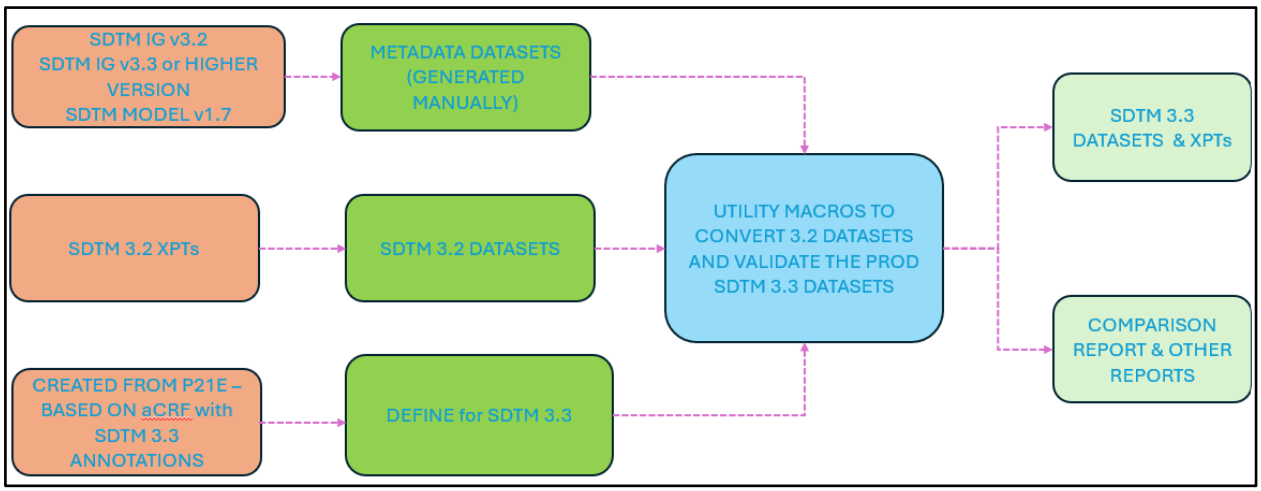


*Figure 2. Programming Flow of Up - Versioning*

## PRODUCTION CONVERSION PROCESS

The development of metadata datasets based on SDTM IG v3.2, SDTM IG 3.3, and SDTM model v1.7 details/files is the initial stage of the conversion process. To find disparities between the properties of SDTM 3.2 datasets and SDTM IG v3.3 metadata datasets, a gap analysis report is generated in the Excel format. To find updates such as variable labels, domain labels, variables to be added, variables to be removed, and supp variables to parent variables, intermediate datasets are made.

The following utilities are created for seamless conversion of SDTM 3.2 datasets to SDTM 3.3 datasets

- ✓ Utility to convert SDTM 3.2 XPTs to datasets, if the XPTs are shared for the conversion
- ✓ Utility to update ARMNRS and ACTARMUD Variables in DM as per SDTMIG 3.3 update.
- ✓ Utility to change the TS domain in accordance with SDTM IG 3.3. The following TSPARMCDs have been updated: SDTMIGVER, SDTMVER, TSVAL have been split into TSVAL1 and TSVAL2, and RANDOM TSPARMCD has been added.
- ✓ Utility to derive xxLOBXFL for the findings domains.
- ✓ Utility to map new domains based on the SDTM IG 3.3. For example, the creation of AG domain based on the

data collected

- ✓ Utility to expand the usage of the established domains by converting the existing mapping, which is discontinued in SDTM IG 3.3. Mapping the SDTM 3.2 CC dataset to the SDTM 3.3 RS dataset, for instance.

```
%macro util_generic_transform(dsn=,dsn1=,CAT=);
%if %length(&dsn.) ne 0 %then %do;
proc sort data=output.&dsn. out= &dsn._1;
by &dsn.TESTCD;
run;
proc contents data=&dsn._1 out=_td_cont(keep=varnum name);
run;
proc contents data=&dsn._1 out=_td_cont1 noprint;
run;
proc sql noprint;
select name into: var_lst separated by " "
from _td_cont1;
quit;
%put &var_lst;
proc sql noprint;
select count(*) into:count from _td_cont;
select name into :var1-:var%cmpres(&count) from _td_cont;
quit;
%let _temp =;
%let word_cnt=%sysfunc(countw(&var_lst));
%do i=1 %to &word_cnt.;
%let _wrd = %scan(&var_lst, &i, " ");
%if %upcase(%substr(&_wrd,1,2)) eq &dsn. %then %do;
%let _wrd = &dsn1.%substr(&_wrd,3);
%end;
%let _temp = &_temp &_wrd;
%end;
data work.&dsn1.(rename = (
%do i=1 %to &count;
%if(%substr(_&&var&i,1,3)=_&dsn.) %then %do;
&dsn.%substr(&&var&i,3)=&dsn1.%substr(&&var&i,3)
%end;
%end;));
set &dsn._1;
Domain="&dsn1.";
run;
data output.&dsn1.;
set %if %sysfunc(exist(output.&dsn1.)) %then output.&dsn1.;
work.&dsn1.;;
run;
proc sort data=meta.sdtmig_33_metadata_variables out=Sdtmig_meta;
by domain seq_no;
run;
proc sql;
create table varorder as select a.* from Sdtmig_meta(where=(domain="&dsn1."))
a where strip(a.domain)||strip(a.Variable_Name) in (select
strip(b.memname)||strip(b.name) from sashelp.vcolumn b where libname='OUTPUT'
and memname="&dsn1." );
quit;
proc sql noprint;
select strip(Variable_Name) into:var_list separated by ' '  from varorder;
quit;
data output.&dsn1.;
retain &var_list.;
set output.&dsn1.;
```

```
run;
proc sort data=output.&dsn1.;
by usubjid &dsn1.SEQ;
run;
%put &var_list;
%if %sysfunc(exist(sdtm.supp&dsn.)) %then %do;
data work.supp&dsn1.;
set sdtm.supp&dsn.;
RDOMAIN="&dsn1.";
IDVAR="&dsn1.SEQ";
run;
data output.supp&dsn1.;
set %if %sysfunc(exist(output.supp&dsn1.)) %then output.supp&dsn1.;
work.supp&dsn1.;;
run;
%end;
proc datasets library=output;
delete %if %sysfunc(exist(output.&dsn.)) %then &dsn.;
%if %sysfunc(exist(output.supp&dsn.)) %then supp&dsn.;;
run;
%end;
%mend util_generic_transform;
```

- ✓ Utility to determine which variables belong to SUPPxx as per SDTM IG 3.2 and which should be mapped into the parent domain in accordance with SDTMIG 3.3 in the parent domains, QNAM is a variable. For instance, CMRSDISC was included as a parent domain variable in SDTM IG 3.3 despite not being a component of the parent domain in SDTM IG 3.2.

```
%macro util_supp_parent_domain;
proc sql noprint;
select distinct memname into : supp separated by " " from dictionary.columns
where libname eq "SDTM" and memname like "SUPP%" and name eq "QNAM";
quit;
%put &supp;
%macro suppvar;
%let count=%sysfunc(countw(&supp.));
%do i=1 %to &count.;
proc sql noprint;
create table stsup&i. as select distinct qnam as variable_name, rdomain as
domain
from sdtm.%scan(&supp.,&i.);
quit;
%end;
data Final;
length variable_name $80.;
length domain $50.;
set stsup:;
   run;
%mend suppvar;
%suppvar;
proc sql;
create table sup_chk as select a.Domain , b.variable_name as supp_variable
from meta.SDTMIG_33_METADATA_VARIABLES a left join work.Final b
on a.domain=b.domain and a.variable_name=b.variable_name
where supp_variable ne " ";
quit;
proc sql noprint;
select domain,supp_variable, count(supp_variable) into:dom
separated by " ",:suppvar separated by " ", :count_suppvar from sup_chk;
```

```
quit;
%put &dom,&suppvar,&count_suppvar;;
%if %length(&suppvar.) %then %do;
%macro merge_supp;
%let dom_count=%sysfunc(countw(&dom.));
%do i=1 %to &dom_count;
%let db=%scan(&dom.,&i.);
data supp&db.;
set sdtm.supp&db.;
&db.SEQ=input(idvarval,best.);
run;
proc sort data=supp&db.;
by studyid usubjid &db.seq;
run;
proc transpose data=supp&db. out=supp&db._t(drop=_:);
by studyid usubjid &db.seq;
id qnam;
idlabel qlabel;
var qval;
where qnam="&suppvar";
run;
proc sql;
select distinct domain_label into :ulabel from meta.sdtmig_33_metadata_datasets
where domain_name eq "&db.";
quit;
%put &ulabel;
data output.&db.(label=&ulabel.);
merge sdtm.&db.(in=a) supp&db._t;
by studyid usubjid &db.seq;
if a;
run;
data output.supp&db.(drop=&db.seq);
set supp&db.;
if qnam="&suppvar" then delete;
run;
proc sort data=OUTPUT.supp&db.;
by  studyid rdomain usubjid idvar idvarval qnam;
run;
%end;
%mend merge_supp;
%merge_supp;
%end;
%mend util_supp_parent_domain;
```

- ✓ Utility to update the domain specific attributes in SDTM 3.3 datasets. For example, SDTM 3.3 CM dataset label to be updated as “Concomitant/Prior Medications” from “Concomitant Medications”
- ✓ Utility to add variables into domains based on GAP Analysis Report generated through utility and any expected variable that is part of domain as per SDTM IG 3.3.
- ✓ Utility to handle the CORE changes from REQ/EXP to PERM from SDTM IG 3.2 to SDTM IG 3.3 and newly added variables like xxLOBXFL. If these PERM variables are not collected on CRF and are missing throughout, then these variables will be dropped from the final SDTM 3.3 dataset.
- ✓ Utility to retain the correct variable order as per SDTM IG 3.3.
- ✓ Utility to update the variable labels/ Domain labels as per SDTMIG 3.3.
- ✓ Utility to transform updated SDTMIG 3.3 Dataset to SAS XPT’s for the P21 validation purposes.

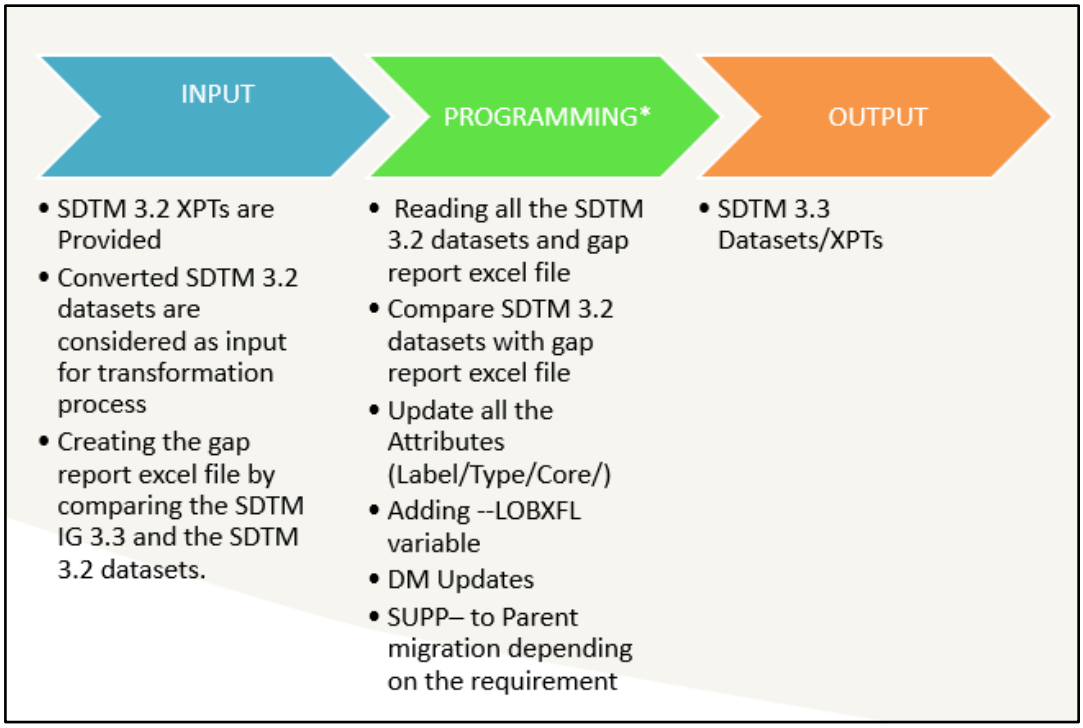


*Figure 3. PROD Programming Flow*

## QC OF THE CONVERSION

It is advised to adapt double programming process to achieve utmost quality of transformed SDTM 3.3 datasets. The following utilities are created for the creation of QC SDTM 3.3 datasets for the comparison process.

- Gap Analysis report is created to identify the differences between attributes of SDTM3.2 datasets and SDTM IG 3.3 variables. Also to compare the gap analysis report from the production to make sure the differences identified are similar and appropriate.
- Utility to update ARMNRS and ACTARMUD Variables in DM as per SDTMIG 3.3 update. To make sure value for the ACTARMUD is updated when ARMCD ne ACTARMCD

```
%macro util_val_armnrs_update;
data output.dm;
set output.dm;
length ARMNRS ACTARMUD $50;
if ARMCD in("SCRNFAIL","NOTASSGN") then do;
if ARMCD="SCRNFAIL" then ARMNRS="SCREEN FAILURE";
else if armcd="NOTASSGN" then ARMNRS="NOT ASSIGNED";
ARMCD ="";
ARM="";
ACTARMCD="";
ACTARM="";
ACTARMUD="";
end;
else if ACTARMCD="NOTTRT" then do;
ACTARMCD="";
ACTARM="";
ARMNRS="ASSIGNED, NOT TREATED";
ACTARMUD="";
end;
else if ARM ne ACTARM then do;
ARMNRS="";
ACTARMUD="";/******** confirm values from client************/
```

```
end;
else do;
ARMNRS="";
ACTARMUD="";
end;
run;
%mend util_val_armnrs_update;
```

- Utility to derive xxLOBXFL variable in all finding domains. Based on Key variables and for all xxBLFL present SDTM 3.2 datasets.
- Utility to transform any decommissioned domain (e.g.:CC) to SDTM 3.3 domain (e.g.: RS). SDTM 3.3 domain can be decided by looking into aCRF page/ discussion with stakeholders.
- Utility to identify variables that are required to move into parent domain as per SDTMIG 3.3.
- Utility to add variables into domains based on gap analysis report created by the validation and any expected variable that is part of domain as per SDTM IG 3.3.
- Utility to handle PERM variables in SDTM 3.3 datasets
- Utility to update/add various attributes of the SDTM 3.2 datasets based on the SDTM IG 3.3 metadata.
- Utility to compare PROD and validation SDTM 3.3 datasets. The comparison report to created in PDF format with all details and stored in the documentation folder for further reviews

```
%macro util_val_compare_datasets(base_lib=, compare_lib=);
proc sql noprint;
select distinct memname,count(memname) into: domain_name separated by " " ,
:cnt trimmed from dictionary.tables where libname='PROD';
quit;
%put &domain_name,&cnt;
ods pdf file="&path/Dataset_compare_report.pdf";
%do i=1 %to &cnt;
%let domain=%scan(&domain_name,&i);
proc compare base=&base_lib..&domain compare=&compare_lib..&domain;
run;
%end;
ods pdf close;
%mend util_val_compare_datasets;
```

- Utility for comparing the PROD SDTM 3.3 and SDTM 3.2 datasets to ensure that no observations have changed and that only metadata has been updated. Datasets from SDTM 3.2 and SDTM 3.3 will be compared in terms of common domains. The detailed comparison report must be produced in PDF format and saved in the documentation folder for further reviews.

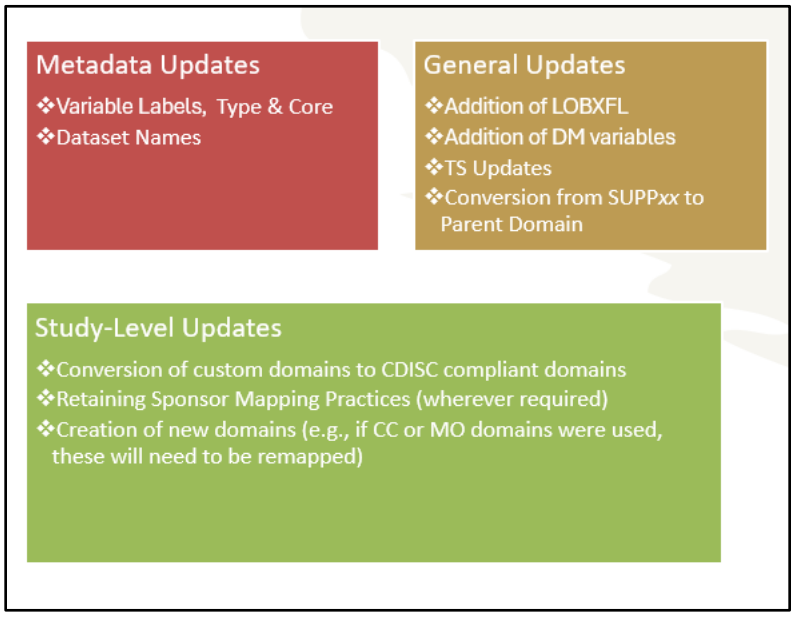


*Figure 4. VAL Programming Logic*

Apart from the above double programming comparison explained above manual QC is done on the reverse approach method. By reading the gap analysis report and all the SDTM 3.3 datasets and making sure that whether the SDTM IG 3.3 updates are incorporated and no gaps in the transformation and reported in granular level report in excel file.

## CONFORMANCE CHECK

P21E report will be analyzed for any metadata issues and annotating the P21E report with explanations for the errors/warnings. When performing validation of SDTM v3.3 datasets using, it is advised you use the SDTM and SDTMIG Conformance Rules v2.0 Rules for.

## SDTM UP-VERSIONING – DO'S

- Keeping up with the CDISC evolution
- Determine the key updates and changes in SDTM IG 3.2 to the higher versions of SDTM IG.
- Bear in mind the processes in place for requesting new domains or changes to the standard
- Should always verify the most recent model, IG, and non-standard variable registry
- Should check with the appropriate foundational teams to determine impact.
- Preparing your team for a smooth transformation process and implementation
- Gather necessary study data and documentation as pre-requisite
- Maintain high standards for data quality and compliance.
- Perform quality assurance checks post-transformation

## SDTM UP-VERSIONING – DONT'S

- Correcting inaccuracies in SDTMIG 3.2 mapping that are not associated with updates to the SDTM version.
- Address the data issues in the SDTM 3.2 datasets
- Refrain from creating or updating sponsor-specific domains that do not adhere to CDISC standards.

## CONCLUSION

We have presented some of our practical experiences and insights from the process of transforming SDTM 3.2 datasets into SDTM 3.3 datasets. We trust that these best practices and resources will aid others in successfully upgrading their SDTM 3.2 datasets to the necessary higher IG versions. This paper includes efficient strategies and programming workflows for addressing many of the significant and nuanced changes that come with these versions.

## REFERENCES

Soumya Rajesh, CSG Llc. – an IQVIA Business. Your Guide to Successfully Upversioning CDISC Standards - https://pharmasug.org/proceedings/2024/DS/PharmaSUG-2024-DS-276.pdf

Lauren Shinaberry, Managing CDISC version changes: how & when to implement? - https://www.cdisc.org/system/files/all/Education/Webinar%2020141002%20-%20Managing%20CDISC%20Version%20Changes.pdf

## ACKNOWLEDGMENTS

We would like to express our warm gratitude to the management of the organization – "Ephicacy" for the encouragement and support in helping with all the necessary facilities to write this paper.
Our sincere thanks to our families and team for their unwavering support and encouragement.

## RECOMMENDED READING

SDTM IG 3.3

SDTM v1.7

SDTM and SDTMIG Conformance Rules v2.0

## CONTACT INFORMATION

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

Author Name: Utkarsh Pathak
Company: Ephicacy Lifescience Analytics
Address: No.6, 2nd Floor, 2nd Main Rd, Arekere,
Off. Bannerghatta Road, City, State ZIP: Bangalore, Karnataka 560076
Work Phone: +91 - 7838558382
Email: utkarsh.pathak@ephicacy.com
Website: www.ephicacy.com

Author Name: Charumathy Sreeraman
Company: Ephicacy Canada Inc.
Address: 2 Robert Speck Parkway, Suite 700, Mississauga, Ontario, L4Z 1H8
Work Phone: +1 – (647) 887 - 9942
Email: charumathy.sreeraman@ephicacy.com
Website: www.ephicacy.com