PharmaSUG 2020 - Paper SS-317

Improving the Quality of Define.xml: A Comprehensive Checklist Before Submission

Ji Qi, Yan Li, and Lixin Gao, Biopier Inc.

ABSTRACT

The define.xml is the cover letter in Module 5 of the electronic Common Technical Document (eCTD) submission to the U.S. Food and Drug Administration (FDA) which provides a high-level summary of the metadata for all the data submitted. A functioning, complete and informative define.xml is required by FDA regulation. A high-quality define.xml will not only aid in the ease of FDA review but also convey the attentiveness of the sponsor's work attitude to the reviewers and augment their trust in the results. In this paper, we will provide a list of details to check and fix before submission of the define.xml, focusing on those for clinical tabulation data (SDTM) and analysis data (ADaM), based on a review of common issues reported in papers as well as our own experience generating define.xml for clients.

INTRODUCTION

Under section 745A(a) of the Federal Food, Drug, and Cosmetic Act (FD&C Act), at least 24 months after the issuance of a final guidance document in which the Food and Drug Administration (FDA or Agency) has specified the electronic format for submitting certain submission types to the Agency, such content must be submitted electronically and in the format specified by FDA[1]. Following this regulation, studies starting on or after December 17, 2016 are required to submit standardized study data using FDA-supported data standards located in the FDA Data Standards Catalog[2] for New Drug Applications (NDA), Abbreviated New Drug Applications (ANDA), and certain Biologics License Applications (BLA) submissions to Office of Medical Products and Tobacco, Center for Drug Evaluation and Research (CDER) and Office of Medical Products and Tobacco, Center for Biologics Evaluation and Research (CBER)[3]. For a clinical study submission, both a define.xml package for SDTM data and a define package for ADaM data should be placed in Module 5 of the eCTD[4,5]. "Define.xml is a required component for both packages to inform the regulators which datasets, variables, controlled terms, and other specified metadata were used"[6]. It uses Extensible Mark-up Language (XML) to specify a set of rules for encoding documents in a format that is both human-readable and machine-readable. Besides that, a study data reviewer guide (cSDRG) or an analysis data reviewer's guide (ADRG) in Portable Document Format (PDF) is not required but recommended. These documents facilitate reviewers to get familiar and navigate through the submitted data. For SDTM packages, the annotated case report forms (aCRF) are also required to provide information on how data was collected. All data submitted electronically in the packages are in the SAS Transport Format (XPORT) version 5 (with an extension of .xpt)[4]. For FDA standards, regulations and guidance documents relating to clinical study electronic submissions, please refer to the recommended reading section.

In this paper, we present a proposed process for define package quality control in our company, with an itemized checklist we recently developed. For some bullet points, we provide in separate paragraphs discussions such as examples, FDA guidance or the check/fix procedure for the relevant point. The procedure and checklist are developed based on a review of common issues in the literature as well as our own experience. We are open to discussions with colleagues in the field for improvements.

QUALITY CONTROL PROCESS FOR THE DEFINE PACKAGE

For our submission package projects, we propose the following process for quality control. Guided by the itemized checklist detailed in the next section, each define package will go through three rounds of review after creation: first by the production person, then by a review person. When items on the checklist have all been checked twice, a senior person in the team will perform a final round of review and sign to accept that the package is submission-ready and deliverable to the clients.

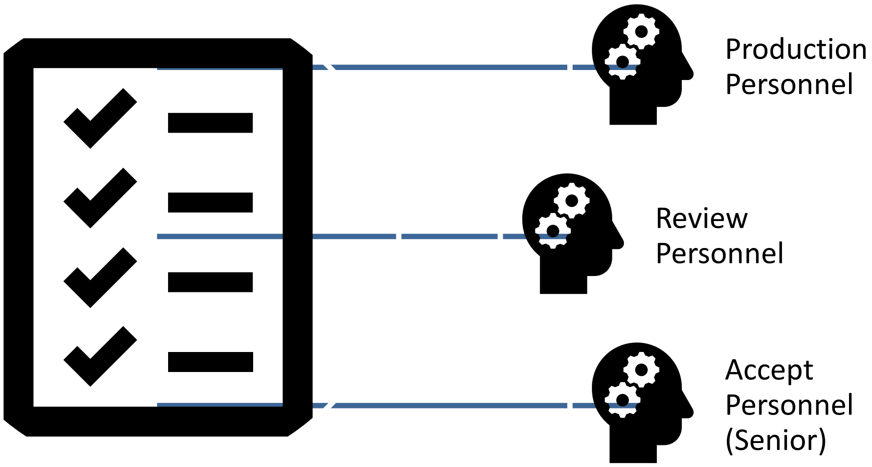

Figure 1. Graphic Illustration for the Define Package Quality Control Process

ITEMIZED CHECKLIST FOR THE SDTM DEFINE PACKAGE

CHECK THE PRESENCE OF ALL ESSENTIAL COMPONENTS

- Check the presence of all SDTM datasets including the trial design domains in .xpt format. Data with no records should not be submitted. For in-house prepared datasets, check timestamp to make sure the latest version is in the submission folder.
- Check the presence of aCRF with the correct file name acrf.pdf.

  Previously, this document used the name blankcrf.pdf but now it should all be changed to acrf.pdf[3]*.*
- Check the presence of the study data reviewer's guide (csdrg.pdf).
- Check the presence of define.xml and define.pdf.
- Check the presence of the stylesheet define2-0-0.xsl.
- Check the presence of openCDISC validation check reports for the define.xml itself, as well as for data validated together with define.xml.

CHECK THE READABILITY OF THE DOCUMENTS

- Make sure the define.xml can be opened in internet browsers with a readable content.

  Several reasons can lead to problems reading the define.xml: missing stylesheet, mis-referencing of the stylesheet in define.xml, an outdated stylesheet (which can lead to a define.xml to be readable but with non-functioning navigation pane)[7]*, hidden special characters, or other syntax errors in define.xml. The latter two issues which are not readily detectable in regular browser view can be identified and fixed if we open the 'inspect element box' with the F12 key.*
- Make sure the define.xml used v2.0.

  In the FDA Data Standards Catalog, two versions of Define.xml are supported: v1.0 and v2.0. But the support for Define-XML v1.0 ended on March 15th, 2018[2]*. Define.xml v2.0 is the current recommended version to use. In May 2019, the final version of Define.xml v2.1 was released, with all the information in define.xml v2.0 plus new changes*[7]*. This newest development has not been implemented yet though and is out of the scope of this paper as such.*

- Check the functionality of bookmarks and hyperlinks. Make sure they are not only clickable but also direct readers to the correct files, pages, and sections.

  One of the examples that usually cause an issue in this category is the aCRF. When SDTM specification was created, certain CRF page numbers were put as the ORIGIN for variables. But later when the aCRF was created, since only unique CRF pages rather than the full document were included, page numbers change. If the page number reference in the ORIGIN column does not update accordingly, the links will direct the readers to a wrong or even non-existing page.

  In our practice, for SDTM variables from CRF, page numbers were not included when SDTM specification was built. After CRF annotation was done. We will read the aCRF comments, export the information of domain, variable name, and CRF page numbers to an excel file, and merge with the specification. This way, all the page numbers would be up to date. In the meantime, if any variable from CRF does not get a number after the merging, we will know immediately that we have issues in CRF annotation that needs to be fixed. More about CRF annotation will be discussed later.

  Another scenario that causes a problem in the links is when we copy a section of content from Protocol, SAP or specification. Any links that are copied over referencing a not submitted file then become an 'orphan link' and need to be fixed.

ACRF CONTENT CHECK

- Make sure two sets of bookmarks are created: one by CRF pages as collected, one by the domain names.
- Check that all variables in SDTM specification with ORIGIN as CRF have corresponding page numbers. If not, check the following two points.
- Make sure domain and variable name spellings are correct in the annotation.
- Make sure all fields are annotated. Fields that are collected but not submitted should be mapped as 'NOT SUBMITTED'.

  Most of the missing annotations happen in SUPPQUAL domains due to last-minute changes in mapping specs[9].

VALIDATION REPORTS CHECK

- Check validation reports timestamps to make sure validations are run on the latest version of datasets and define.xml. If any changes are made, validation checks need re-run.
- Check the engine, configuration and Controlled Terminology (CT) version used in the validation checks. The latest versions should be used unless otherwise specified in SAP or requested by the regulatory agency.
- Go through every issue found in the validation reports. Fix all that is fixable in programming.

CSDRG CONTENT CHECK

- Make sure the latest version of the Pharmaceutical Users Software Exchange (PHUSE) cSDRG template is used and the corresponding completion guide is followed. Currently, we should use v1.2 2015-01-26[10].

  For both cSDRG and ADRG, templates are not specified in FDA documents, but the templates from PHUSE are widely accepted in the industry.

- Check that all sections of the template are completed.
- Check that the aCRF annotation conventions are explained with examples in cSDRG.
- Check that we do not miss explanations for sponsor defined rules.
- Check that an appendix is provided for the full-text inclusion/exclusion criteria from all versions of

protocols.

- Check that the number of issues listed in the validation check issue summary section matches the number in the actual report.
- Check that all validation check issues are clearly explained.

  Different scenarios of issues require different levels of explanation. Several good papers discussed the proper explanation for conformance issues for SDTM and /or ADAM datasets.[3,11,12] *One issue has been specially pointed out in the Technical Specifications Document - Study Data Technical Conformance Guide from FDA*[4]*: the use of Controlled Term 'OTHER'. Mapping a collected value to 'OTHER' is not recommended when there are controlled terms available to match the collected value, regardless of whether the controlled terminology, also called codelist, is extensible. Each value in –TERM mapped to a –DECOD value of 'OTHER' calls for an explanation.*

- Grammar check.
- Check for consistency of tense, voice, point of view, font and size, table formatting[3].
- After all the above points are checked and fixed, double-check the acronyms list. Make sure it covers all acronyms that are used in the document and delete any that are not used.
- When checking acronyms, also make sure the full spelling is provided the first time an acronym appears in the text.
- Lastly, make sure the table of contents is updated.

DEFINE.XML CONTENT CHECK

- Make sure the essential sections are all present: tabulation datasets, codelists, value-level metadata and external dictionaries.
- Avoid using surrogate keys (such as -SEQ) as the key variables for datasets[9,13]. Use natural keys instead, such as AETERM, AESTDTC etc.
- Computational algorithms should be provided for all variables with ORIGIN as 'Derived' or 'Assigned'.
- No raw data or other non-submitted data should be referenced in the computational algorithms.
- The algorithm should be explained in plain language rather than complex code.

  Since the audience of the define.xml package includes non-programmers, we could not simply copy codes into the algorithm. But if snippets of codes are necessary, they could be included in addition to the textual explanations.

- The text in the algorithm should provide the right amount of detail.

  Overall, the goal of the computational algorithm section is to provide detailed and reproducible logics for all the derived and assigned variables in a clear but concise manner.

- For variables without standardized codelist, sponsor defined coldelist should be provided to help the reviewers understand how the data was collected.
- For the codelist 'UNIT', do not merge values from all domains into one codelist, instead, create separate unit codelist for each domain that has this value[7].
- A codelist should contain all possible values for the variables assigned to it.

  If the ORIGIN for the variable is CRF or protocol, then all values from the document should be included, no matter how many values are in the actual data. For variables from other ORIGIN, the codelist is composed of distinct values from the actual data.

- For Clinical Data Interchange Standards Consortium (CDISC) standard codelists, there is no need to include all values in each study-specific define.xml file since some of the codelists can be extensive, such as COUNTRY, LBTEST, etc. Only the values relevant to the study would be included.

- Only include codelists applicable to the current define package. Do not include codelists intended for other packages such as the ADaM define package[14].
- For codelists with coding and decoding, check the one-to-one mapping based on SDTMIG requirement[14].

  For example, VISIT and VISITNUM should map one-to-one for all scheduled visits.
- Check character case since define.xml is case sensitive.

  For CDISC standard codelists, use upper case values. Lower case values will result in OpenCDISC check warning messages.

CROSS CHECKS AMONG DOCUMENTS

- Check that the number of submitted datasets match the number described in the documents.
- Make sure the metadata presented in the documents, including dataset labels, key variables, variable name and labels matched the actual data.
- Make sure the duplicated information, including protocol number and title, data standards versions, medical dictionary version, and drug dictionary version, are consistent across documents.
- Make sure the data standard version and codelist version in the documents matches what is actually used in the validation.

ITEMIZED CHECKLIST FOR THE ADAM DEFINE PACKAGE - DIFFERENCE FROM THE SDTM DEFINE PACKAGE

For ADaM define review, we will follow similar steps with SDTM Define package review. To avoid duplication and to focus on important points, only the differences between the two will be tabulated here.

CHECK THE PRESENCE OF ALL ESSENTIAL COMPONENTS

- ADaM define packages do not have the aCRF component.
- In addition to ADaM datasets, ADaM programs as well as Table/Figure/Listing reports programs need to be submitted.
- The reviewer guide in ADaM define package is named adrg.pdf.

CHECK THE READABILITY OF THE DOCUMENTS

No major differences with SDTM define packages to be discussed.

VALIDATION REPORTS CHECK

No major differences with SDTM define packages to be discussed.

ADRG CONTENT CHECK

- Make sure the latest version of the PHUSE ADRG template is used and the corresponding completion guide is followed. Currently, we should use v1.1 2015-01-26[16].
- In ADRG, there is a section about the source data used for the creation of the analysis dataset. If any data besides SDTM are used, they should be provided in the Appendix section.

  For example, a table is usually used as a reference for the determination of adverse events of interest (AESI). In another case, CDC or WHO growth datasets are commonly used for the calculations of weight, height, BMI and head circumference percentiles and z-scores. Since the growth datasets are big, links to the online resources are provided in ADRG instead.
- Check that all study-specific or sponsor defined derivation rules or variable conventions have been explained in ADRG.

Due to the analysis nature of ADaM datasets, the creation process for them requires far more derivation for the variables compared with that for the SDTM datasets. Hence, it is more critical for ADRG to provide details to guide the reviewers' understanding. Common rules for visit windowing, treatment mapping, and date imputation, etc. are put into section 3. Subjects who required special analysis rules are also put into section 3. Special variable conventions are put into section 4. Dataset specific derivation rules are put into section 5. For each ADaM dataset that has special rules to be explained, a separate section would be used. The corresponding sections are also commonly referenced in section 3 when general rules are discussed.

- Make sure that all variable computational algorithms that are referred to ADRG for details are properly documented here (see define.xml content check section below).
- Any dataset that is 5GB in size or larger needs to be split for submission purpose[4]. That needs to be documented in ADRG.

DEFINE.XML CONTENT CHECK

- Computational algorithm should be explained in plain language rather than complex code.
- For codelists with coding and decoding, check the one-to-one mapping based on ADAMIG requirement[17].

  For ADaM datasets, the one-to-one mapping rule applies more broadly than in SDTM datasets. Hence this checking becomes more important. For example, AVISIT and AVISITN should map one-to-one, PARAM and PARAMCD should map one-to-one, and within each parameter, AVAL and AVALC should map one-to-one for non-missing AVALC

- Check that no computational algorithm column contains texts that are too lengthy.

  For ADaM datasets, especially in Integrated Summary of Safety (ISS) or Integrated Summary of Efficacy (ISE) studies, certain variable derivations could be very complicated. Since formatting such as numbering or bullet points are not supported in define.xml, lengthy texts will not be reader-friendly. In our practice, when the computational algorithm cannot be explained within 1000 characters, we will move the details to ADRG where formatting can be applied to improve the clarity of the text and provide a reference link in define.xml.

CROSS CHECKS AMONG DOCUMENTS

- Check that the number of submitted datasets match the number described in the documents.
- Make sure the metadata presented in the documents, including dataset labels, key variables, variable name and labels matched the actual data.
- Make sure the duplicated information, including protocol number and title, data standards versions, medical dictionary version, and drug dictionary version, are consistent across documents.
- Make sure the data standard version and codelist version in the documents matches what is actually used in the validation.

ALIGNMENT BETWEEN SDTM AND ADAM DEFINE PACKAGES

- All the information that is included in both packages should be consistent.
- All the cross-documents consistency checks performed within each package should also be performed between the two packages.
- It is not required but recommended to have similar language utilization styles between the two packages.

CONCLUSION

To ensure the quality of the define.xml, triple check for items in the above list should be performed before submission. Moreover, all the regulations, standards, specifications, guidance and templates documents referenced in this paper are constantly being revised (for example, when we started drafting this paper, we read the FDA *Technical Specifications Document* - *Study Data Technical Conformance Guide* v4.4, but when it is time to finalize the reference list, we realize that the document has v4.5 released in March 2020). Therefore, it is important to keep our knowledge of the field up to date and always use the versions of standards supported in the FDA Data Standards Catalog.

REFERENCES

[1] FDA, "Providing Regulatory Submissions in Electronic Format — Standardized Study Data". Available at https://www.fda.gov/media/82716/download.

[2] FDA, "Data Standards Catalog". Available at https://www.fda.gov/industry/fda-resources-data-standards/study-data-standards-resources.

[3] PHUSE, "Best Practices for Documenting Dataset Metadata: Define-XML versus Reviewer's Guide". Available at: https://www.phusewiki.org/docs/Deliverables/Best%20Practices%20for%20Documenting%20%20Dataset%20Metadata%20-%20Define-XML%20versus%20%20Reviewers%20Guide%20-%2005APR2019.pdf .

[4] FDA, "Technical Specifications Document - Study Data Technical Conformance Guide". Available at https://www.fda.gov/media/136460/download.

[5] FDA, "Providing Regulatory Submissions in Electronic Format — Certain Human Pharmaceutical Product Applications and Related Submissions Using the eCTD Specifications". Available at https://www.fda.gov/media/135373/download.

[6] CDISC, "Define-XML" Accessed April 5, 2020. https://www.cdisc.org/standards/data-exchange/define-xml.

[7] Glass at al., "Common Define.xml File Issues Seen During FDA's JumpStart Service" PhUSE US Connect 2018, Poster PP11. Available at: https://www.lexjansen.com/phuse-us/2018/pp/PP11_ppt.pdf.

[8] CDISC, "Define-XML v2.1" Accessed April 6, 2020. Available at: https://www.cdisc.org/standards/foundational/define-xml/define-xml-v21.

[9] Sirichenko et al., "What is high quality study metadata?" PharmaSUG 2016, Paper SS11. Available at: https://www.lexjansen.com/pharmasug/2016/SS/PharmaSUG-2016-SS11.pdf.

[10] PHUSE, "Study Data Reviewer's Guide" Accessed April 6, 2020. Available at: https://www.phusewiki.org/wiki/index.php?title=Study_Data_Reviewer%27s_Guide.

[11] Kelly K., "Best Practice for Explaining Validation Results in the Study Data Reviewer's Guide" PharmaSUG 2018, Paper SS13. Available at: https://www.lexjansen.com/pharmasug/2018/SS/PharmaSUG-2018-SS13.pdf.

[12] Moore G. E., "A Practical Guide to the Issues Summary in the Data Conformance Summary of Reviewer's Guides" PharmaSUG 2019, Paper SS092. Available at: https://www.lexjansen.com/pharmasug/2019/SS/PharmaSUG-2019-SS-092.pdf.

[13] Sirichenko S., "How to use SUPPQUAL for specifying natural key variables in define.xml?" PharmaSUG 2019, Paper DS318. Available at: https://www.lexjansen.com/pharmasug/2019/SS/PharmaSUG-2019-SS-318.pdf.

[14] Roustone D., "Do's and Don'ts of Define.xml" PHUSE EU Connect, Paper SA04. Available at: https://www.lexjansen.com/phuse/2018/sa/SA04.pdf.

[15] CDISC, "SDTMIG" Accessed April 6, 2020. https://www.cdisc.org/standards/foundational/sdtmig

[16] PHUSE, "Analysis Data Reviewer's Guide" Accessed April 6, 2020. Available at: https://www.phusewiki.org/wiki/index.php?title=Analysis_Data_Reviewer%27s_Guide.

[17] CDISC, “ADAMIG” Accessed April 6, 2020. https://www.cdisc.org/standards/foundational/adam/adam-implementation-guide-v11.

RECOMMENDED READING

- *Guidance to Industry – Providing Regulatory Submissions in Electronic Format: Standardized Study Data*
- *Technical Specifications Document - Study Data Technical Conformance Guide*
- *Study Data Standards Resources Web Page*
- *FDA Data Standards Catalog*
- *FDA Portable Document Format Specifications*
- *Specifications for File Format Types Using eCTD Specifications*
- *Guidance to Industry – Providing Regulatory Submissions in Electronic Format: Submissions Under Section 745A(a) of the Federal Food, Drug, and Cosmetic Act*
- *Guidance to Industry – Providing Regulatory Submissions in Electronic Format: Certain Human Pharmaceutical Product Applications and Related Submissions Using the Electronic Common Technical Document Specifications*

CONTACT INFORMATION

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

Ji Qi
Biopier Inc.
jiqi@biopier.com

Yan Li
Biopier Inc.
yli@biopier.com

Lixin Gao
Biopier Inc.
lgao@biopier.com