Paper SM16

ADRG Builder Python Application

Pierre Dostie, UCB Pharma S.A., Colombes, France

ABSTRACT

In the world of statistical programming, the Analysis Data Reviewer's Guide often remains overlooked when it comes to automation and standardization. However, there are several sections within the ADRG where standardization and automation could significantly impact both the quality and consistency. Consider elements from Define.xml to be replicated within ADRG, serving as a starting point for automating creation. We've developed a Python™ application that offers users with a range of options, from building an initial version of the ADRG document to producing a delivery ready PDF file. Users can also capitalize on the standardization aspect of the tool by integrating standard analysis datasets subsection text, ensuring consistency across trials and projects. Integration of validation tool reports into the ADRG is also an automated process handled by this tool. Finally, users have access to reports throughout the ADRG development process, aiding them in identifying remaining issues and offering solutions.

INTRODUCTION

Creating an Analysis Datasets Reviewers Guide (ADRG) is often regarded as a daunting task by authors of this document. The challenges in producing a valid and meaningful document are numerous. Frequently, it is created at the last minute, resulting in a quality that does not always meet industry standards. Although the PHUSE working group has provided a template and guidelines, errors and inconsistencies can still appear in the final version of the document.

Imagine a tool that simplifies the creation of an ADRG, automatically filling out sections from the Define.xml file it complements. This tool could also generate standard content for standard analysis datasets and highlight in a check report any areas where information is missing or inaccurate. Additionally, it could automatically import Pinnacle 21 Enterprise™ validation reports into the ADRG (1). Such a tool would greatly enhance the efficiency and accuracy of the ADRG creation process, ensuring higher quality and consistency.

The development of tools is part of a long-term strategy of standardization within UCB's Statistical Programming department. In this paper, we will demonstrate how a standardization process can be applied to the creation of an ADRG. We will outline the development of a Python tool designed specifically for this purpose. We will detail the process from designing a template file to producing a delivery ready ADRG file, showing how this tool integrates into our standardization strategy.

The choice of Python as a technology for the development of tools in the pharmaceutical industry is widely documented and is not the topic of this paper.

ADRG PURPOSE

According to the latest release of the Study Data Technical Conformance Guide from the FDA, "*The preparation of an Analysis Data Reviewer's Guide (ADRG) is recommended as an important part of a standards-compliant analysis data submission for clinical trials.*" (2). The main purpose of an ADRG is to "*provide FDA reviewers with context for analysis datasets and terminology, received as part of a regulatory product submission, additional to what is presented within the data folder (i.e., define.xml).*" It should also "*provide a summary of ADaM conformance findings.*". The ADRG complements the ADaM define.xml file with information that can't either be described properly in a define.xml (i.e. long or complex derivations/algorithm) or that cannot be represented in that format (i.e. conformance findings, etc..). The ADRG Builder tool ensures the connection between a define.xml file and an ADRG is maintained and helps identify potential inconsistencies.

ADRG TEMPLATE AND CONFIGURATION FILE

The ADRG Package version 1.2 released by the PHUSE on July 18th, 2019, is the de facto standard used in the industry as of today (3). It contains, on top of the ADRG template itself, a completion guideline as well as numerous example files. A summary of changes from the previous version is also included in this package. We used this ADRG template as a starting point to develop our own customized template ADRG file. We performed the following changes

to the template:

- Removed unnecessary text / information.
- Added “<<” and “>>” tags to identify areas of interest.
- Reconfigured some tables headers and format.

In addition to creating a customized ADRG template file, we created a JSON configuration file (Figure 1) to regulate the structure and contents of the ADRG generated by the application. We used the ADRG Completion Guidelines to establish the rules for ADRG compliance. For example, mandatory sections or mandatory text within sections are monitored with the application using the JSON configuration file. Also included are Microsoft Word® styles (i.e. Heading 1, Heading2, etc.…) each section header should use to ensure consistent formatting throughout the document to facilitate the creation of a table of contents.


```
[
    {
        "number": "1.",
        "regex": "^Introduction$",
        "style": "Heading 1",
        "empty": "Yes"
    },
    {
        "number": "1.1",
        "regex": "^Purpose$",
        "style": "Heading 2",
        "empty": "No",
        "text": "^This document provides context for the analysis datasets and terminology that benefit from additional explanation beyond the Data Definition document \\(define\\.xml\\) for an individual study\\. In
addition, this document provides a summary of ADaM conformance findings\\."
    },
    {
        "number": "1.2",
        "regex": "^Acronyms$",
        "style": "Heading 2",
        "empty": "No",
        "acronyms": ["ADaM","ADRG","IG","NA","SDTM","TAUG"]
    },
    {
        "number": "1.3",
        "regex": "^Study Data Standards and Dictionary Inventory$",
        "style": "Heading 2",
        "empty": "No",
        "standards": ["SDTM","SDTM Controlled Terminology","ADaM","ADaM Controlled Terminology","Data Definitions","TAUG (if applicable)","Medications Dictionary","Medical Events Dictionary","Other standards (optional)"]
    },
```


Figure 1

ADRG BUILDER PYTHON APP

A desire to improve both the automation and standardization processes in Statistical Programming at UCB led to the development of multiple in-house tools over the last few years (SAS® code generator, TFLs templates, Define.xml, etc..). The ADRG Builder Python application was the next logical step in this ongoing effort. The goals of the application are:

- Maintain a high level of quality and consistency for ADRGs created at UCB.
- Reduce time spent typing/copying text by automating the creation of sections/sub-sections.
- Include derivation text for some standard analysis datasets developed by UCB ADaM Standards Team.
- Fully automate the insertion of a P21 Enterprise report in a dedicated ADRG section.
- Generate check report that highlights remaining issues and potentially provide solutions.

It is worth noting that the application doesn’t fully automate the creation of ADRGs and that authors still need to fill out most of the contents. Still, having some of the work already done and standardized makes authors’ life much easier.

We assessed the lifecycle of an ADRG creation and came up with 4 tasks the application had to perform:

1. Create a first ADRG version.
2. Update an existing ADRG.
3. Create final version of an ADRG.
4. Create a delivery ready PDF version of an ADRG.

Figure 2 shows how those tasks appear when a user launches the application.

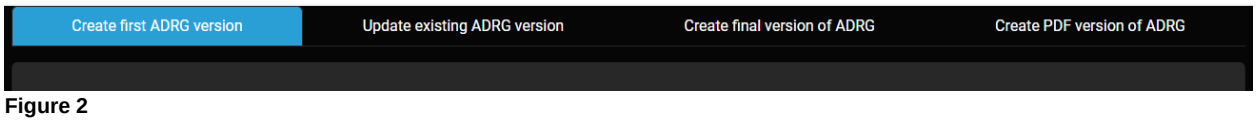


Figure 2

The following sections provide a detailed explanation of each task, including the required input files, the files generated, and the updates made to the ADRG contents upon each task completion.

FIRST ADRG VERSION

The timing for creating an ADRG document varies based on authors' preferences. Typically, the ADRG is developed in parallel with the ADaM define.xml it is designed to complement. It is rarely completed in a single attempt, often resulting in multiple versions. Regardless, an initial version of the ADRG must be created at some stage. To do so, users select the tab "Create first ADRG version" located on the left-hand side at the top of the screen (Figure 3). This tab is selected by default when you open the application.

Users select a compliant define.xml file (either 2.0 or 2.1) by clicking the associated "Browse" button. An error message is triggered if the file is not compliant with their corresponding define.xml schema (4). If the selected file is compliant, the list of analysis dataset(s) available in define.xml appears as a list of checkboxes. Users can unselect analysis dataset(s) they deemed not ready to be added in the first version of the ADRG.

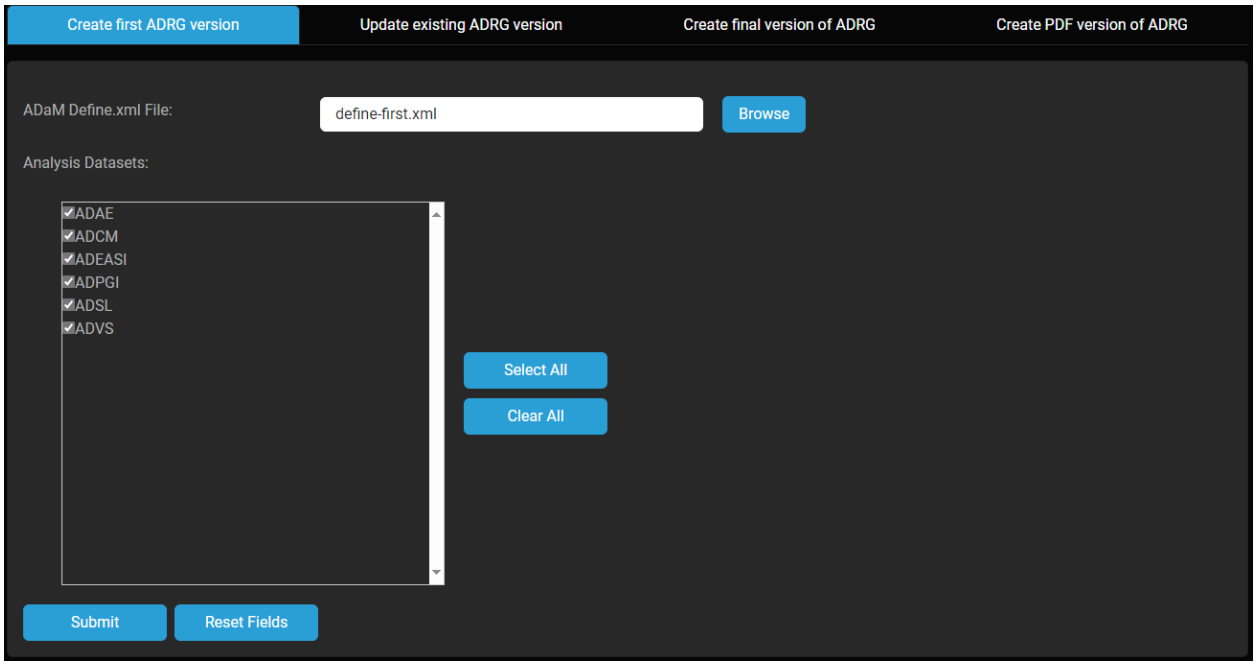


Figure 3

Once users are happy with their choices, they click the "Submit" button and the process of generating a first version of the ADRG starts. If no errors occur, then a pop-up window appears saying the first version of the ADRG is created. Users also see a Microsoft Word® file called "adrg-yyyy-mm-dd-hh-mm.docx" being downloaded.

The following sections highlight each element that is updated from the original ADRG template to create the initial version of the ADRG.

COVER PAGE

The cover page of our customized ADRG template file contains a tagged field "<<Protocol Number>>" (Figure 4).

Analysis Data Reviewer's Guide

UCB

Study <<Protocol Number>>

Figure 4

This tagged field is replaced with the value from StudyName element in define.xml as shown below (Figure 5).

```
<Study OID="Study-ABC123">
   <GlobalVariables>
      <StudyName>ABC123</StudyName>
      <StudyDescription>PHASE 2, MULTIPLE DOSE, BLINDED STUDY OF DRUG XXX</StudyDescription>
      <ProtocolName>ABC123</ProtocolName>
   </GlobalVariables>
```

UCB
Study ABC123

Figure 5

SECTION 1.3

Section 1.3 table from the ADRG template contains all mandatory rows required by the ADRG Completion Guidelines document (Figure 6).

1.3 Study Data Standards and Dictionary Inventory

Standard or Dictionary	Versions Used
SDTM	
SDTM Controlled Terminology	
ADaM	
ADaM Controlled Terminology	
Data Definitions	
TAUG (if applicable)	
Medications Dictionary	
Medical Events Dictionary	Initial: Final:
Other standards (optional)	

Figure 6

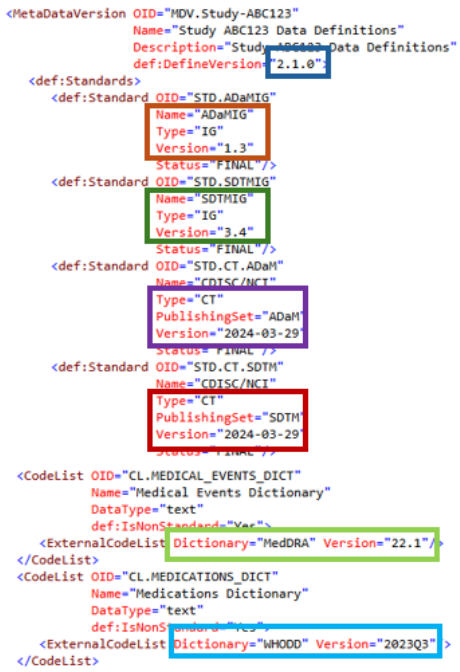
Values for "Version Used" column come from multiple define.xml elements (Figure 8).

- Data Definitions value come DefineVersion attribute's value of MetaDataVersion element.
- SDTM and ADaM Controlled Terminology values come from Standards elements (define.xml 2.1 only).
- SDTM and ADaM Implementation Guides value(s) come from Standards elements (define.xml 2.1 only) and are cross referenced against a look-up JSON file to extract the full document name(s) (Figure 7).

```
{
    "ADaMIG": {
        "1.3" : [
            {
            "document": "ADaM Model Document Version 2.1"
            },
            {
            "document": "ADaM Implementation Guide Version 1.3"
            },
            {
            "document": "ADaM Structure for Occurrence Data (OCCDS) Implementation Guide Version 1.1",
            "class": ["OCCURRENCE DATA STRUCTURE"]
            },
            {...},
            {...}
        ],
        "1.2" : [...],
        "1.1" : [...]
    },
    "SDTMIG": {
        "3.4" : [
            {
            "document": "Study Data Tabulation Model Version 2.0"
            },
            {
            "document": "Study Data Tabulation Model Implementation Guide: Human Clinical Trials Version 3.4"
            }
        ],
        "3.3" : [...],
        "3.2" : [...]
    },
    "SDTMIG-AP": {
        "1.0" : [
            {
            "document": "Study Data Tabulation Model Implementation Guide: Associated Persons Version 1.0"
            }
        ]
    },
    "SDTMIG-MD": {
        "1.1" : [
            {
            "document": "Study Data Tabulation Model Implementation Guide for Medical Devices Version 1.1"
            }
        ],
        "1.0" : [...]
    }
}
```

Figure 7

The application also checks each analysis datasets classes (and subclasses if applicable) and adds specific implementation guide(s). For example, in Figure 8 shown below, since there is at least one analysis dataset in the define.xml that is of class “Occurrence Data Structure” (ADAE), the “ADaM Structure for Occurrence Data (OCCDS) Implementation Guide Version 1.1” is automatically added to the list of documents for ADaM.



1.3 Study Data Standards and Dictionary Inventory

Standard or Dictionary	Versions Used
SDTM	Study Data Tabulation Model Version 2.0 Study Data Tabulation Model Implementation Guide: Human Clinical Trials Version 3.4
SDTM Controlled Terminology	2024-03-29
ADaM	ADaM Model Document Version 2.1 ADaM Implementation Guide Version 1.3 ADaM Structure for Occurrence Data (OCCDS) Implementation Guide Version 1.1
ADaM Controlled Terminology	2024-03-29
Data Definitions	define.xml v2.1.0
TAUG (if applicable)	
Medications Dictionary	WHODD 2023Q3
Medical Events Dictionary	Initial: Final: MedDRA 22.1
Other standards (optional)	

Figure 8

SECTION 2.1

Section 2.1 of the ADRG template has three tagged fields as shown below (Figure 9).

2.1 Protocol Number and Title

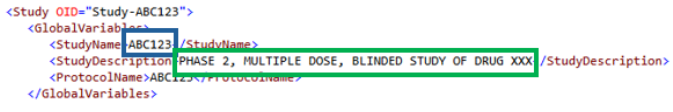
Protocol Number: <<Protocol Number>>

Protocol Title: <<Protocol Title>>

Protocol Versions: <<Protocol Versions>>

Figure 9

Tagged field “<<Protocol Number>>” is replaced with value from StudyName element whereas “<<Protocol Title>>” tagged field is replaced with StudyDescription element value in define.xml (Figure 10).



2.1 Protocol Number and Title

Protocol Number: ABC123

Protocol Title: PHASE 2, MULTIPLE DOSE, BLINDED STUDY OF DRUG XXX

Protocol Versions: <<Protocol Versions>>

Figure 10

SECTION 3.1

Section 3.1 of the ADRG template contains an empty table for “Core Variables” to be added as shown below (Figure 11).

3.1 Core Variables

Core variables are those that are represented across all/most analysis datasets.

Variable Name	Variable Description

Figure 11

"Core Variables" are defined in the ADRG Completion Guideline as "*those that are represented across all/most analysis datasets*". It also states that "*Since both USUBJID and STUDYID are required by the ADaM model, then at a minimum, this table would contain these two variables.*".

The application applies an algorithm to extract what is considered "Core Variables". Authors of the ADRG are responsible to agree or not with the algorithm and to add/remove items from the table if deemed necessary. Authors also need to add "Variable Description" values to complete the table. It is important to note that the algorithm relies solely on the ADaM define.xml contents, assuming the analysis datasets are structured accordingly.

An example of how the table would be populated for column "Variable Name" after the application has run is shown in the figure below (Figure 12).

3.1 Core Variables

Core variables are those that are represented across all/most analysis datasets.

Variable Name	Variable Description
STUDYID	
USUBJID	
BMIBLGR1	
TRT01P	
TRTSEQP	
TRTSEQA	

Figure 12

SECTION 5.2

Section 5.2 of the ADRG template contains an empty table for analysis datasets in the study, starting with ADSL and followed with others in alphabetical order (Figure 13).

5.2 Analysis Datasets

Dataset Dataset Label	Class	Efficacy	Safety	Baseline or other subject characteristics	PK/PD	Primary Objective	Structure

Figure 13

The application extracts the following information from each ItemGroupDef element in the define.xml file (Figure 14):

- Dataset comes from Name attribute's value
- Dataset Label comes from Description/TranslatedText element value
- Class comes from Class element value
- Structure comes from Structure attribute's value

```
<ItemGroupDef OID="ADSL"
              Name="ADSL"
              Purpose="Analysis"
              SASDatasetName="ADSL"
              Repeating="No"
              IsReferenceData="No"
              def:Structure="One record per subject"
              def:ArchiveLocationID="LF.ADSL"
              def:CommentOID="COM.ADSL"
              def:StandardOID="STD.ADaMIG">
   <Description>
      <TranslatedText>Subject-Level Analysis Dataset</TranslatedText>
   </Description>
   <ItemRef .../>
   <def:Class Name="SUBJECT LEVEL ANALYSIS DATASET" />
   <def:leaf ID="LF.ADSL" xlink:href="adsl.xpt">
      <def:title>adsl.xpt</def:title>
   </def:leaf>
</ItemGroupDef>
```

5.2 Analysis Datasets

Dataset Dataset Label	Class	Efficacy	Safety	Baseline or other subject characteristics	PK/PD	Primary Objective	Structure
ADSL Subject-Level Analysis Dataset	ADSL						One record per subject
ADAE Adverse Events Analysis Dataset	OCCDS						One record per adverse event per subject
ADCM Concomitant Medications Analysis Dataset	OCCDS						One record per recorded medication occurrence or constant-dosing interval per subject
ADEASI EASI Questionnaire Analysis Dataset	BDS						One record per Eczema Area and Severity Index (EASI) question per visit per subject
ADPGI PGI Analysis Dataset	BDS						One record per question per visit per subject
ADVS Vital Signs Analysis Dataset	BDS						One record per vital sign measurement per visit per date and time per timepoint per subject

Figure 14

SECTIONS 5.2.X

Sections 5.2.X are created by the application starting with “5.2.1 ADSL - Subject-Level Analysis Dataset”. A new section is created for each other analysis dataset. The application contains a library of standard content developed by UCB ADaM Standards Team, allowing analysis dataset sections to be automatically populated if the relevant dataset exists in the library. This process ensures consistency and accuracy across all ADRGs. If a standard content doesn’t exist for an analysis dataset, a tagged value “<<insert your text here>>” is added to the section instead.

5.2.1 ADSL - Subject-Level Analysis Dataset

ADSL contains variables to support baseline characteristics and disposition analyses. The population indicator variables and covariates are defined in ADSL and copied into other analysis datasets as needed.

<< All subjects in domain DM, except for screen failures were included in ADSL. >>

5.2.2 ADAE - Adverse Events Analysis Dataset

ADAE contains variables to adverse events analyses.

<< All rows in domain AE were included in ADAE. >>

5.2.3 ADCM - Concomitant Medications Analysis Dataset

<<insert your text here>>

5.2.4 ADEASI - EASI Questionnaire Analysis Dataset

ADEASI is an Eczema Area and Severity Index (EASI) questionnaire dataset that supports the *<<Primary/Secondary/Exploratory>>* efficacy endpoint of the study.

The analysis EASI total score is derived as follow:

- For each Body Region, create a temporary score:
  - Create a temporary sum of AVAL from the following 4 assessments (see parameters referenced in table below): Erythema, Edema/Papulation, Excoriation and Lichenification
  - Multiply the temporary calculated above with the associated Region score value (AVAL from associated parameter referenced in table below).
  - Multiply the value from the previous step with the value from column “Multiplier” of the associated Body Region (see reference table below).
- Sum all 4 Body Region temporary score calculated above to get the EASI total score. The minimum score is 0 and the maximum score is 72.
- Missing values are imputed based on <<XXX>> method.

Patients 8 years old or above:

Body Region	Erythema (0-3)	Edema/ Papulation (0-3)	Excoriation (0-3)	Lichenification (0-3)	Region score (0-6)	Multiplier	Score per body region
Head / neck	PARAMCD= “EASI0102”	PARAMCD= “EASI0103”	PARAMCD= “EASI0104”	PARAMCD= “EASI0105”	PARAMCD= “EASI0101	0.1	(0-7.2)
Trunk	PARAMCD= “EASI0107”	PARAMCD= “EASI0108”	PARAMCD= “EASI0109”	PARAMCD= “EASI0110”	PARAMCD= “EASI0106	0.3	(0-21.6)
Upper extremities	PARAMCD= “EASI0112”	PARAMCD= “EASI0113”	PARAMCD= “EASI0114”	PARAMCD= “EASI0115”	PARAMCD= “EASI0111	0.2	(0-14.4)
Lower extremities	PARAMCD= “EASI0117”	PARAMCD= “EASI0118”	PARAMCD= “EASI0119”	PARAMCD= “EASI0120”	PARAMCD= “EASI0116	0.4	(0-28.8)
					The final EASI score is the sum of the 4 region scores:		(0-72)

Figure 15

In the example above (Figure 15), ADSL, ADAE and ADEASI have standard contents coming from a library of standards. ADCM has no standard content, and a tagged text is generated by the application. Tagged text can also be part of standard contents as demonstrated in ADSL, ADAE and ADEASI sections.

UPDATE EXISTING ADRG VERSION

Upon the creation of a first ADRG version, users can manually update sections/tables not covered by the application. It is advisable to follow the recommendations from the ADRG Completion Guideline version 1.2 when updating the contents of an ADRG file. It is not mandatory to fill out all missing/incomplete sections/tables of an ADRG file before proceeding with the second option of the application, “Update of an existing ADRG version” (Figure 16).

This task could be performed as many times as required. Each execution generates both an updated ADRG as well as a check report file. The check report file (covered in a dedicated section of this paper) highlights issues, inconsistencies, and missing pieces of information in the ADRG file.

To proceed with this task, users need to select the ADRG file they want to update by clicking the associated “Browse” button. Similarly, users need to select the latest version of the ADaM define.xml. This could be the same file used for creating the first version of the ADRG or an updated file. As in the previous task, users can unselect analysis dataset(s) they deem not ready to be added in the ADRG.

Optionally, users can add a Pinnacle 21 Enterprise report file by clicking the associated “Browse” button.

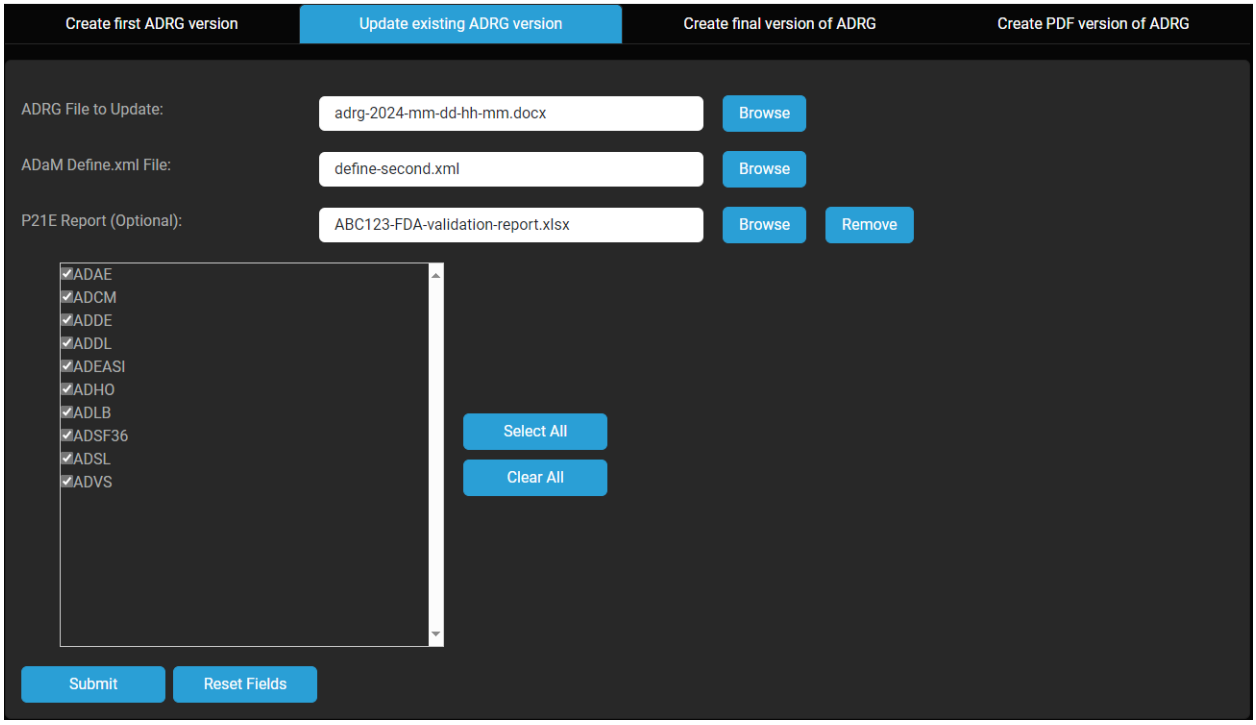


Figure 16

Users click the “Submit” button and the process of generating an updated version of the ADRG starts. If one or more analysis dataset(s) exist in section 5.2 table of the ADRG file and is **not** present in the ADaM define.xml file, then the application generates a pop-up window such as the following (Figure 17):

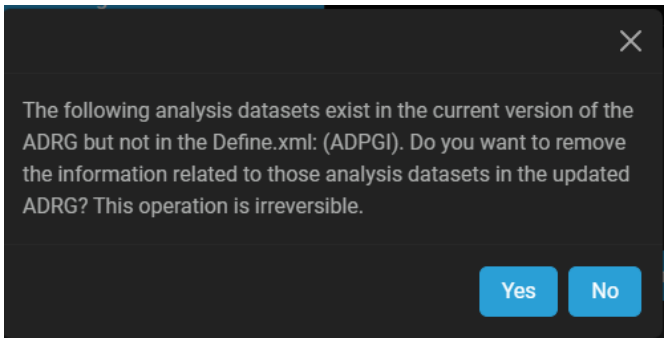


Figure 17

In the example above, the application has detected that analysis dataset ADPGI exist in section 5.2 table of the ADRG file but not in the ADaM define.xml. Users can choose if they either want to remove this analysis dataset (ADPGI) in the ADRG or not. By clicking “Yes”, the analysis dataset is removed from section 5.2 table and the associated 5.2.x section is removed as well. Otherwise, all information related to this analysis dataset is kept in the new ADRG.

A pop-up window appears saying the ADRG file was updated and to check the report to fix the remaining issues if there are any (Figure 18).

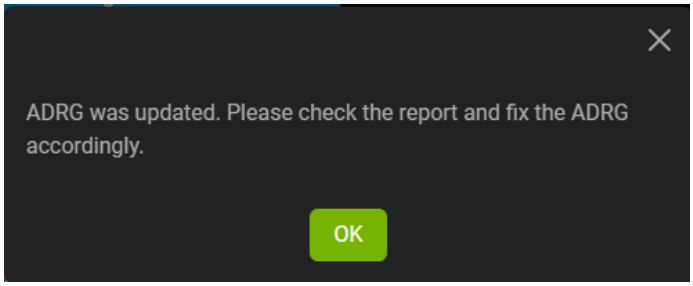


Figure 18

A zip file called “adrg-yyyy-mm-dd-hh-mm.zip” is automatically downloaded. It contains both the updated version of the ADRG file as well as a check report file in Microsoft Excel® format. The following sections illustrate each item updated in the ADRG file from this task.

COVER PAGE

An update like the one described earlier is performed only if the StudyName element in the define.xml differs from the value in the ADRG file. Otherwise, no update is performed on the cover page.

SECTION 1.3

Table in section 1.3 is updated if elements in define.xml differs from the ones in the ADRG file to update. For example, if the standard version of a controlled terminology, implementation guide or dictionary (i.e. MedDRA, WHODD) has changed in the define.xml, then it would be reflected in section 1.3 table accordingly.

Similarly, if a new SDTM standard (i.e. Medical Device) is added to the define.xml as well as device related analysis dataset(s), those additions would be reflected in section 1.3 table. The example below shows such a scenario (Figure 19):

```
<def:Standards>
   <def:Standard OID="STD.ADaMIG"
                 Name="ADaMIG"
                 Type="IG"
                 Version="1.3"
                 Status="FINAL"/>
   <def:Standard OID="STD.SDTMIG"
                 Name="SDTMIG"
                 Type="IG"
                 Version="3.4"
                 Status="FINAL"/>
   <def:Standard OID="STD.SDTMIG-MD"
                 Name="SDTMIG-MD"
                 Type="IG"
                 Version="1.1"
                 Status="FINAL"/>

<CodeList OID="CL.MEDICAL_EVENTS_DICT"
          Name="Medical Events Dictionary"
          DataType="text"
          def:IsNonStandard="Yes">
   <ExternalCodeList Dictionary="MedDRA" Version="23.1"/>
</CodeList>
<CodeList OID="CL.MEDICATIONS_DICT"
          Name="Medications Dictionary"
          DataType="text"
          def:IsNonStandard="Yes">
   <ExternalCodeList Dictionary="WHODD" Version="2024Q1"/>
</CodeList>

<ItemGroupDef OID="ADDL"
              Name="ADDL"
              Purpose="Analysis"
              SASDatasetName="ADDL"
              Repeating="No"
              IsReferenceData="No"
              def:Structure="One record per device per subject"
              def:ArchiveLocationID="LF.ADDL"
              def:CommentOID="COM.ADDL"
              def:StandardOID="STD.ADaMIG">
   <Description>
      <TranslatedText>Device-Level Analysis Dataset</TranslatedText>
   </Description>
   <ItemRef ...>
   <def:Class Name="DEVICE LEVEL ANALYSIS DATASET" />
   <def:leaf ID="LF.ADDL" xlink:href="addl.xpt" />
      <def:title>addl.xpt</def:title>
   </def:leaf>
</ItemGroupDef>
```

1.3 Study Data Standards and Dictionary Inventory

Standard or Dictionary	Versions Used
SDTM	Study Data Tabulation Model Version 2.0 Study Data Tabulation Model Implementation Guide: Human Clinical Trials Version 3.4 Study Data Tabulation Model Implementation Guide for Medical Devices Version 1.1
SDTM Controlled Terminology	2024-03-29
ADaM	ADaM Model Document Version 2.1 ADaM Implementation Guide Version 1.3 ADaM Structure for Occurrence Data (OCCDS) Implementation Guide Version 1.1 Analysis Data Model Implementation Guide for Medical Devices Version 1.0
ADaM Controlled Terminology	2024-03-29
Data Definitions	define.xml v2.1.0
TAUG (if applicable)	
Medications Dictionary	WHODD 2024Q1
Medical Events Dictionary	Initial: Final MedDRA 23.1
Other standards (optional)	

Figure 19

SECTION 2.1

An update is performed only if StudyName element and/or StudyDescription element in define.xml differ from the information in the ADRG file. Otherwise, no updates are performed in section 2.1.

SECTION 3.1

Column “Variable Name” in section 3.1 table is updated using the algorithm described earlier in this paper. Variable(s) in the table that belong exclusively to analysis dataset(s) no longer available in the define.xml are deleted from the table. Values in column “Variable Description” are kept as is.

The following are scenarios that result in section 3.1 table being updated as shown below (Figure 20):

- Row for variable “BMIBLGR1” is removed as it was a variable only present in an analysis dataset that is no longer referenced in the define.xml file.
- Row for variable “SAFFL” is added as it is identified as a “Core Variable” by the algorithm. “Variable Description” column is set to null.
- Other rows remain the same as they are still identified as “Core Variables” and the “Variable Description” column values are carried over.

3.1 Core Variables

Core variables are those that are represented across all/most analysis datasets.

Variable Name	Variable Description
STUDYID	Study Identifier
USUBJID	Unique Subject Identifier
BMIBLGR1	BMI at Baseline Analysis Grou
TRT01P	Planned Treatment for Period 01
TRTSEQP	Planned Sequence of Treatment
TRTSEQA	Actual Sequence of Treatment

3.1 Core Variables

Core variables are those that are represented across all/most analysis datasets.

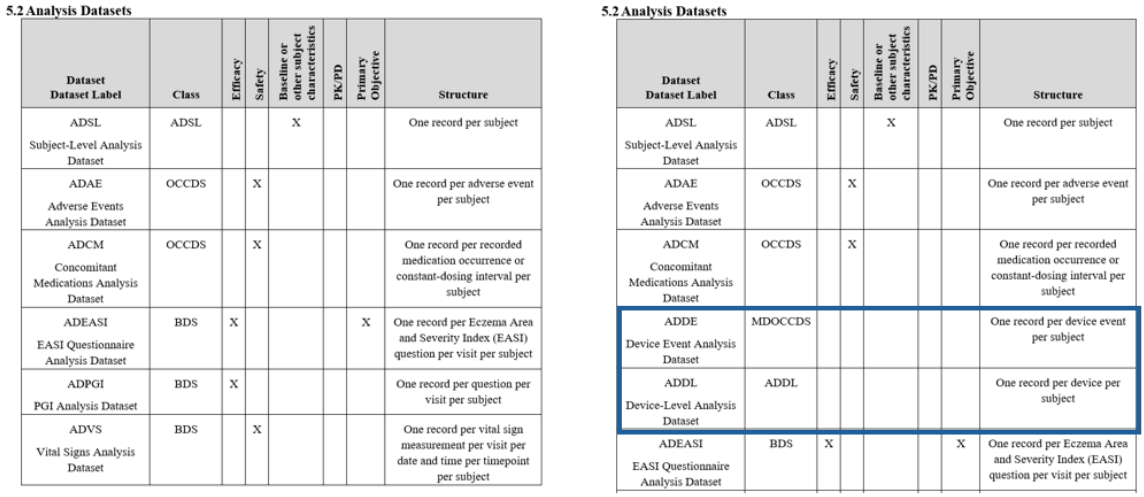
Variable Name	Variable Description
STUDYID	Study Identifier
USUBJID	Unique Subject Identifier
TRT01P	Planned Treatment for Period 01
TRTSEQP	Planned Sequence of Treatment
TRTSEQA	Actual Sequence of Treatment
SAFFL	

Figure 20

SECTION 5.2

Section 5.2 table is updated based on the analysis datasets available in the ADaM define.xml file. Analysis dataset(s) that exist in section 5.2 table that are no longer referenced in define.xml are deleted from the table. Exception to this rule is when users decide otherwise as previously shown (Figure 17). Information computed in columns “Efficacy”, “Safety”, “Baseline or other subject characteristics”, “PK/PD” and “Primary Objective” are carried over. New analysis dataset(s) added to the table would have those columns empty.

The figure (Figure 21) below shows the original table on the left-hand side compared with the table after the application has run the task. Information from the original table is kept whereas new rows are created with empty columns for new analysis dataset(s).



5.2 Analysis Datasets

Dataset Dataset Label	Class	Efficacy	Safety	Baseline or other subject characteristics	PK/PD	Primary Objective	Structure
ADSL Subject-Level Analysis Dataset	ADSL			X			One record per subject
ADAE Adverse Events Analysis Dataset	OCCDS		X				One record per adverse event per subject
ADCM Concomitant Medications Analysis Dataset	OCCDS		X				One record per recorded medication occurrence or constant-dosing interval per subject
ADEASI EASI Questionnaire Analysis Dataset	BDS	X				X	One record per Eczema Area and Severity Index (EASI) question per visit per subject
ADPGI PGI Analysis Dataset	BDS	X					One record per question per visit per subject
ADVS Vital Signs Analysis Dataset	BDS		X				One record per vital sign measurement per visit per date and time per timepoint per subject

5.2 Analysis Datasets

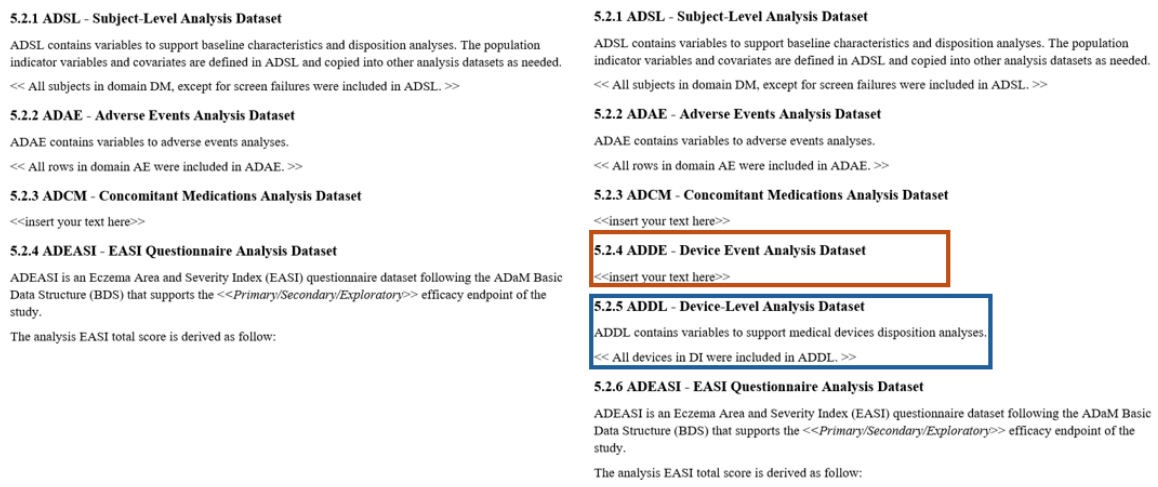
Dataset Dataset Label	Class	Efficacy	Safety	Baseline or other subject characteristics	PK/PD	Primary Objective	Structure
ADSL Subject-Level Analysis Dataset	ADSL			X			One record per subject
ADAE Adverse Events Analysis Dataset	OCCDS		X				One record per adverse event per subject
ADCM Concomitant Medications Analysis Dataset	OCCDS		X				One record per recorded medication occurrence or constant-dosing interval per subject
ADDE Device Event Analysis Dataset	MDOCCDS						One record per device event per subject
ADDL Device-Level Analysis Dataset	ADDL						One record per device per subject
ADEASI EASI Questionnaire Analysis Dataset	BDS	X				X	One record per Eczema Area and Severity Index (EASI) question per visit per subject

Figure 21

SECTIONS 5.2.X

Sections 5.2.x are updated based on the analysis datasets available in the ADaM define.xml file. Sections representing analysis dataset(s) no longer represented in the define.xml are deleted. Exception to this rule is if users decide otherwise as shown in the introduction of this option. Sections numbers are updated to reflect the changes. The contents of each section are carried over. If a new analysis dataset is added and exist in the library of standard contents, then that standard content is added to the dedicated analysis dataset 5.2.x section. Otherwise, a placeholder “<<insert your text here>>” is added to the analysis dataset 5.2.x section.

The figure (Figure 22) below shows an example of original sections on the left-hand side and updated sections on the right-hand side. New sections are created for new analysis datasets (ADDE, ADDL) and sections numbers are updated. ADDL has standard content inserted whereas no such contents exist for ADDE.



5.2.1 ADSL - Subject-Level Analysis Dataset

ADSL contains variables to support baseline characteristics and disposition analyses. The population indicator variables and covariates are defined in ADSL and copied into other analysis datasets as needed.

<< All subjects in domain DM, except for screen failures were included in ADSL. >>

5.2.2 ADAE - Adverse Events Analysis Dataset

ADAE contains variables to adverse events analyses.

<< All rows in domain AE were included in ADAE. >>

5.2.3 ADCM - Concomitant Medications Analysis Dataset

<<insert your text here>>

5.2.4 ADEASI - EASI Questionnaire Analysis Dataset

ADEASI is an Eczema Area and Severity Index (EASI) questionnaire dataset following the ADaM Basic Data Structure (BDS) that supports the <<*Primary/Secondary/Exploratory*>> efficacy endpoint of the study.

The analysis EASI total score is derived as follow:

5.2.1 ADSL - Subject-Level Analysis Dataset

ADSL contains variables to support baseline characteristics and disposition analyses. The population indicator variables and covariates are defined in ADSL and copied into other analysis datasets as needed.

<< All subjects in domain DM, except for screen failures were included in ADSL. >>

5.2.2 ADAE - Adverse Events Analysis Dataset

ADAE contains variables to adverse events analyses.

<< All rows in domain AE were included in ADAE. >>

5.2.3 ADCM - Concomitant Medications Analysis Dataset

<<insert your text here>>

5.2.4 ADDE - Device Event Analysis Dataset

<<insert your text here>>

5.2.5 ADDL - Device-Level Analysis Dataset

ADDL contains variables to support medical devices disposition analyses.

<< All devices in DI were included in ADDL. >>

5.2.6 ADEASI - EASI Questionnaire Analysis Dataset

ADEASI is an Eczema Area and Severity Index (EASI) questionnaire dataset following the ADaM Basic Data Structure (BDS) that supports the <<*Primary/Secondary/Exploratory*>> efficacy endpoint of the study.

The analysis EASI total score is derived as follow:

Figure 22

SECTIONS 6.1 & 6.2

Pinnacle 21 Enterprise (P21E) is the software used at UCB to evaluate the conformance of both analysis datasets and define.xml. Software version and validation engine version information are reported in section 6.1 whereas findings generated from P21E reports are reported in section 6.2. The application can read either FDA or PMDA P21E report files. An error pop-up window is generated if the selected file is not a compliant P21E report.

The figure (Figure 23) below shows how the software version and validation engine version information are extracted from a P21E report and added to section 6.1.

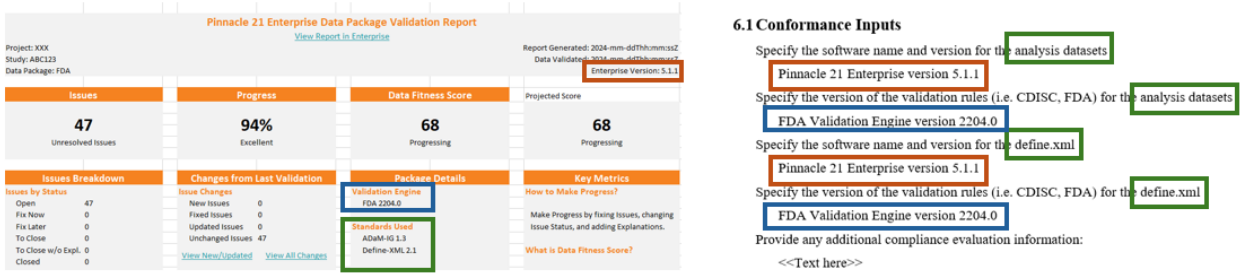


Figure 23

Table in section 6.2 is populated from information in different locations in the P21E report has shown in the figure below. The table is recreated each time a new P21E file report is selected. Explanations need to be added in P21E system rather than in the ADRG. The first two images below represent the locations in the report where the information exist. The last image shows how that information is represented in section 6.2 table (Figure 24).

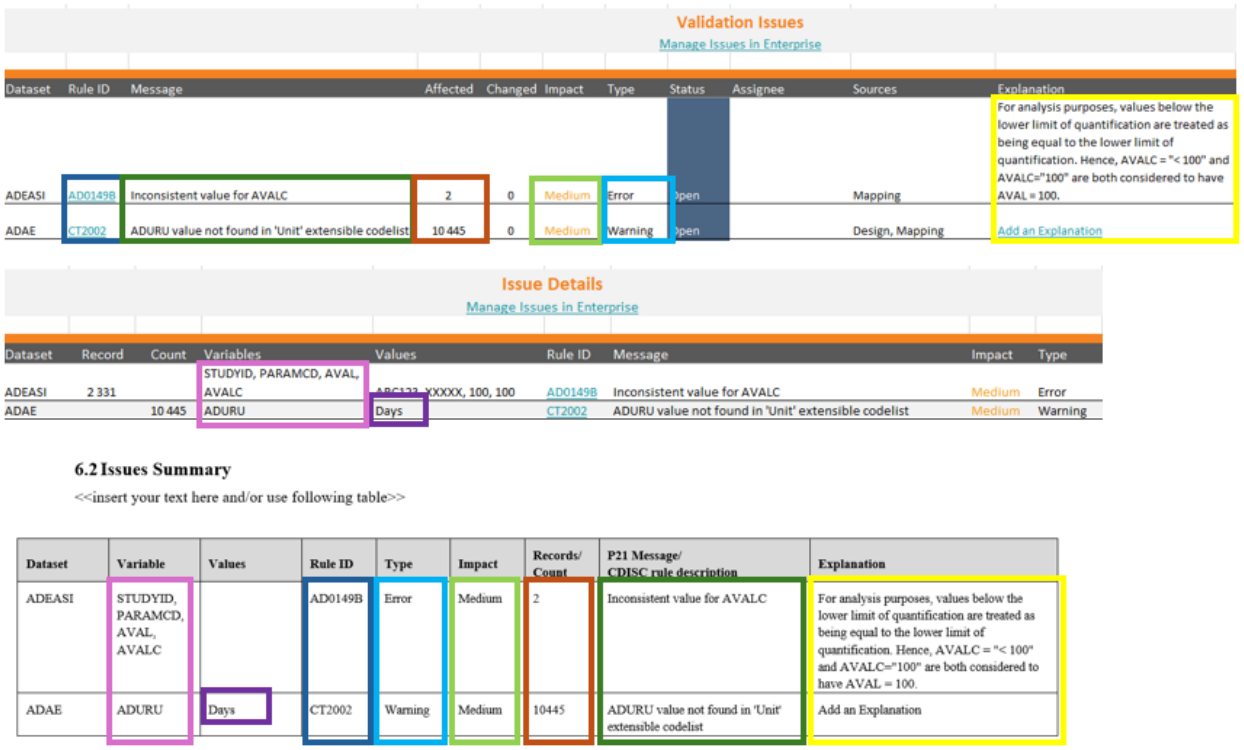


Figure 24

CREATE FINAL VERSION OF ADRG

The third option of the application, “Create Final Version of ADRG”, makes sure that no pending issues remains in the ADRG. It checks the contents of an ADRG against the final version of an ADaM define.xml as well as optionally against a final Pinnacle 21 Enterprise report. If no issues remain, it creates an ADRG ready to be converted into a delivery ready PDF (see “Create PDF Version of ADRG” later in this paper). Otherwise, a check report is created instead with pending issues to be fixed.

To proceed with this task, users need to select the ADRG file they considered as final by clicking the associated “Browse” button. Similarly, users need to select the final version of the ADaM define.xml. Optionally, users can select the final Pinnacle 21 Enterprise report file by clicking the associated “Browse” button (Figure 25).

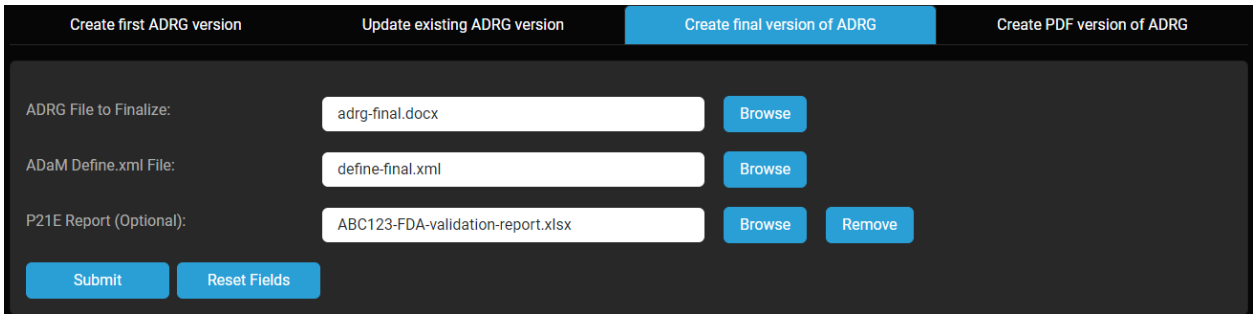


Figure 25

Users click the “Submit” button and the process of creating a final version of an ADRG file starts. It checks if the contents of the ADRG is in adequation with the contents of the ADaM define.xml file and, optionally, with the Pinnacle 21 Enterprise report contents.

If issues remain, the following pop-up window (Figure 26) appears, and a check report is automatically generated by the application. Users need to address the issues highlighted in the check report before proceeding again.

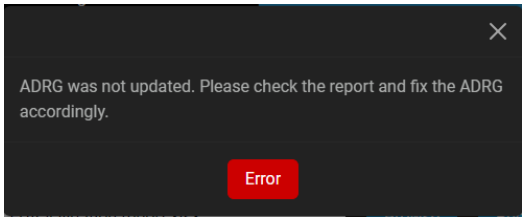


Figure 26

Otherwise, the following pop-up window (Figure 27) appears, and the final ADRG file is generated by the application.

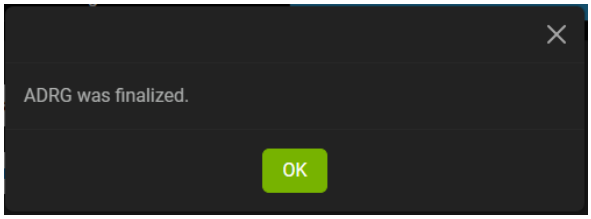


Figure 27

The following operations are performed prior to producing a final ADRG file:

- Removes empty 5.2.x sections and updates remaining 5.2.x sections numbers accordingly.
- Creates hyperlinks in section 5.2 table to their associated 5.2.x sections.

The figure (Figure 28) below shows hyperlinks added to section 5.2 table:

5.2 Analysis Datasets

Dataset Dataset Label	Class	Efficacy	Safety	Baseline or other subject characteristics	PK/PD	Primary Objective	Structure
ADSL Subject-Level Analysis Dataset	ADSL			X			One record per subject
ADAE Adverse Events Analysis Dataset	OCCDS		X				One record per adverse event per subject
ADCM Concomitant Medications Analysis Dataset	OCCDS		X				One record per recorded medication occurrence or constant-dosing interval per subject
ADDE Device Event Analysis Dataset	MDOCCDS		X				One record per device event per subject
ADDL Device-Level Analysis Dataset	ADDL			X			One record per device per subject

Figure 28

The figure (Figure 29) below shows the empty 5.2.x sections removed and renumbered accordingly in the final version:

5.2.1 ADSL - Subject-Level Analysis Dataset

ADSL contains variables to support baseline characteristics and disposition analyses. The population indicator variables and covariates are defined in ADSL and copied into other analysis datasets as needed.

All subjects in domain DM, except for screen failures were included in ADSL.

5.2.2 ADAE - Adverse Events Analysis Dataset

ADAE contains variables to adverse events analyses.

All rows in domain AE were included in ADAE.

5.2.3 ADCM - Concomitant Medications Analysis Dataset

<<insert your text here>>

5.2.4 ADDE - Device Event Analysis Dataset

<<insert your text here>>

5.2.5 ADDL - Device-Level Analysis Dataset

ADDL contains variables to support medical devices disposition analyses.

All devices in DI were included in ADDL.

5.2.1 ADSL - Subject-Level Analysis Dataset

ADSL contains variables to support baseline characteristics and disposition analyses. The population indicator variables and covariates are defined in ADSL and copied into other analysis datasets as needed.

All subjects in domain DM, except for screen failures were included in ADSL.

5.2.2 ADAE - Adverse Events Analysis Dataset

ADAE contains variables to adverse events analyses.

All rows in domain AE were included in ADAE.

5.2.3 ADDL - Device-Level Analysis Dataset

ADDL contains variables to support medical devices disposition analyses.

All devices in DI were included in ADDL.

5.2.4 ADEASI - EASI Questionnaire Analysis Dataset

ADEASI is an Eczema Area and Severity Index (EASI) questionnaire dataset following the ADaM Basic Data Structure (BDS) that supports the Primary efficacy endpoint of the study.

Figure 29

CREATE PDF VERSION OF ADRG

The last option of the application is the creation of a delivery ready ADRG file in PDF format. It applies the finalization instructions from the ADRG Completion Guidelines document as well as some UCB specific standards.

To proceed with this option, users only need to select an ADRG file by clicking the associated “Browse” button (Figure 30).

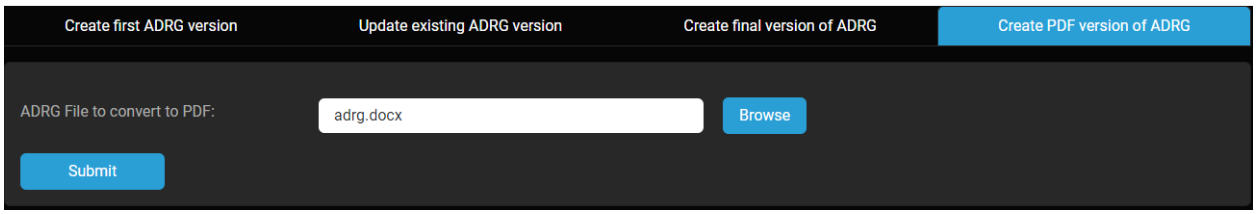


Figure 30

Pressing the “Submit” button users are prompted with pop-up window asking if they updated the table of contents in the ADRG file prior proceeding with the conversion to PDF. The figure (Figure 31) below shows the window in question with instructions on how to update the table of contents if necessary.

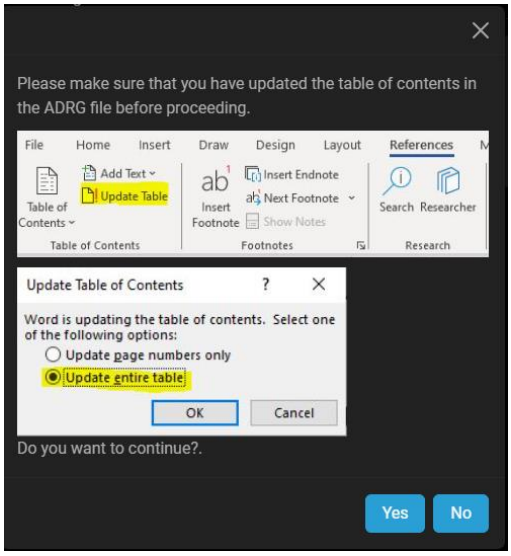


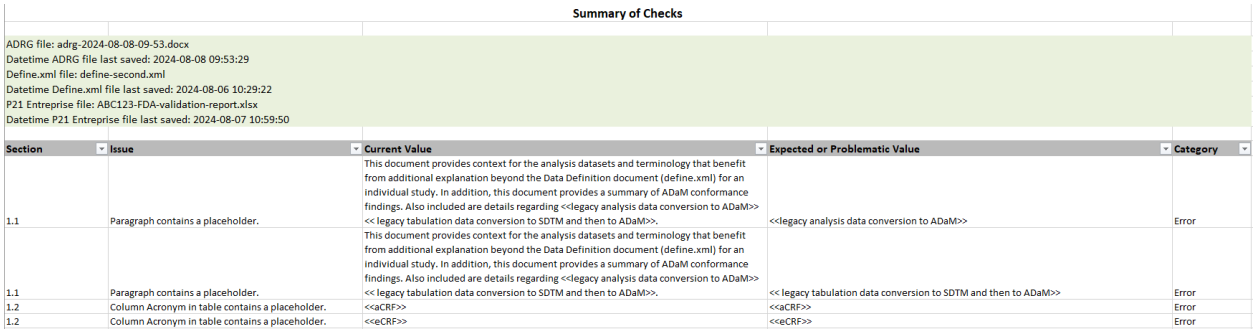
Figure 31

If users click “No”, the process stops, and they can proceed with updating the table of contents of the ADRG file as instructed. Otherwise, if users click “Yes”, a pop-up window appears saying the ADRG in PDF is created. Users also see a PDF file called “adrg-yyyy-mm-dd-hh-mm.pdf” being downloaded.

CHECK REPORT

A check report in Microsoft Excel® format is created when either “Update existing ADRG version” or “Create final version of ADRG” tasks are executed. It helps users to identify remaining issues and contains the following information (Figure 32):

- A header section with both file names and last time they were saved for the ADRG, define.xml and Pinnacle 21 Enterprise report.
- A column “Section” that indicates the section number where the issue can be found.
- A column “Issue” that describes the issue.
- Columns “Current Value” and “Expected or Problematic Value” are optional columns and help users identify issues and potentially provide expected or problematic value.
- A column “Category” that identifies the type of issue, either “Error” or “Note”.



Summary of Checks

ADRG file: adrg-2024-08-08-09-53.docx
Datetime ADRG file last saved: 2024-08-08 09:53:29
Define.xml file: define-second.xml
Datetime Define.xml file last saved: 2024-08-06 10:29:22
P21 Entreprise file: ABC123-FDA-validation-report.xlsx
Datetime P21 Entreprise file last saved: 2024-08-07 10:59:50

Section	Issue	Current Value	Expected or Problematic Value	Category
1.1	Paragraph contains a placeholder.	This document provides context for the analysis datasets and terminology that benefit from additional explanation beyond the Data Definition document (define.xml) for an individual study. In addition, this document provides a summary of ADaM conformance findings. Also included are details regarding <<legacy analysis data conversion to ADaM>> << legacy tabulation data conversion to SDTM and then to ADaM>>.	<<legacy analysis data conversion to ADaM>>	Error
1.1	Paragraph contains a placeholder.	This document provides context for the analysis datasets and terminology that benefit from additional explanation beyond the Data Definition document (define.xml) for an individual study. In addition, this document provides a summary of ADaM conformance findings. Also included are details regarding <<legacy analysis data conversion to ADaM>> << legacy tabulation data conversion to SDTM and then to ADaM>>.	<< legacy tabulation data conversion to SDTM and then to ADaM>>	Error
1.2	Column Acronym in table contains a placeholder.	<<aCRF>>	<<aCRF>>	Error
1.2	Column Acronym in table contains a placeholder.	<<eCRF>>	<<eCRF>>	Error

Figure 32

Checks such as missing/empty or invalid mandatory sections and remaining text placeholders are performed on the entire ADRG document. Checks for specific sections are mostly based on recommendations from the ADRG Completion Guidelines with others being UCB specific. Those checks ensure the ADRG is consistent and compliant with the define.xml file it is supposed to complement.

CONCLUSION

The automation and standardization of the Analysis Data Reviewer’s Guide (ADRG) creation process represent a significant advancement, saving time for authors while ensuring consistency and quality across documents. The

development of the application presented in this paper addresses some of the challenges faced by authors in producing high-quality ADRG documents. It also exemplifies how open-source technology such as Python can improve the overall experience in statistical programming.

REFERENCES

(1) Pinnacle 21 Enterprise™
https://www.pinnacle21.com/enterprise

(2) FDA: Study Data Technical Conformance Guide - Technical Specifications Document.
Study Data Technical Conformance Guide - Technical Specifications Document | FDA

(3) PHUSE: Analysis Data Reviewer's Guide (ADRG) Package
Analysis Data Reviewer's Guide (ADRG) Package - WORKING GROUPS - PHUSE Advance Hub

(4) CDISC®
Define-XML v2.1.7 | CDISC
Define-XML v2.0 | CDISC

ACKNOWLEDGMENTS

The author would like to thank his former manager at UCB, Cindy Stroupe, for her unconditional support.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:
Pierre Dostie
UCB Pharma France
420 rue d'Estienne d'Orves
92700 Colombes, France
Email: pierre.dostie@ucb.com