

Simplify, Automate, Submit

**An In-House Framework for Seamless Metadata Flow from
(Pre)Production to Submission**

Disclaimer: The data and examples shown are partially simulated or modified to illustrate technical concepts. Real-world implementations may vary.

The Journey: From Challenge to Transformation



The Challenge

Define.xml treated as an afterthought

- Inconsistencies and rework
- Quality risks and delays
- No Define-XML 2.1 support



The Vision

Integrated, standardized workflow

- Continuous updates
- Cross-functional collaboration
- Efficiency and traceability



From Vision to Reality

Metadata-driven framework- a living, evolving project

- Agile, modular and scalable by design
- "Do it now" mindset- continuous improvement
- End-to-end traceability



The Result

Reimagined Define.xml & Dataset creation process through

- Automated collaborative framework
- Bridging prescriptive and descriptive metadata through a dynamic, synchronized Define.xml

This transformation is driven by a shift from reactive to proactive metadata management, ensuring reliability, intuitive workflows, and higher quality with minimal manual work.

Driving Modularity: An Integrated Framework Approach

Driver program

Environment Setup

Workflow Orchestration

Multi-user Conflict Prevention

Modular design

Self-contained and purpose-defined

Adhere to coding standards

Plug-and-play expansion



Framework Components

Metadata-driven framework integrating entry specs and data from other sources to automate processing and generate a single, reliable source of truth - a P21-ready Excel output.

Framework Components



Entry File

Define_entry.xlsx

Entry point to Study metadata

Pre-populated P21-style excel template

Optimized for usability

- Reordered columns
- '*Drop*' column for trial-specific exclusions
- Includes special-purpose columns



Processing Tool

P2ISpecCreator

Processes Entry file and data from other sources

Restructures and creates Upload file

Flags easily correctable errors

Saves reusable metadata datasets for other purposes



Upload File

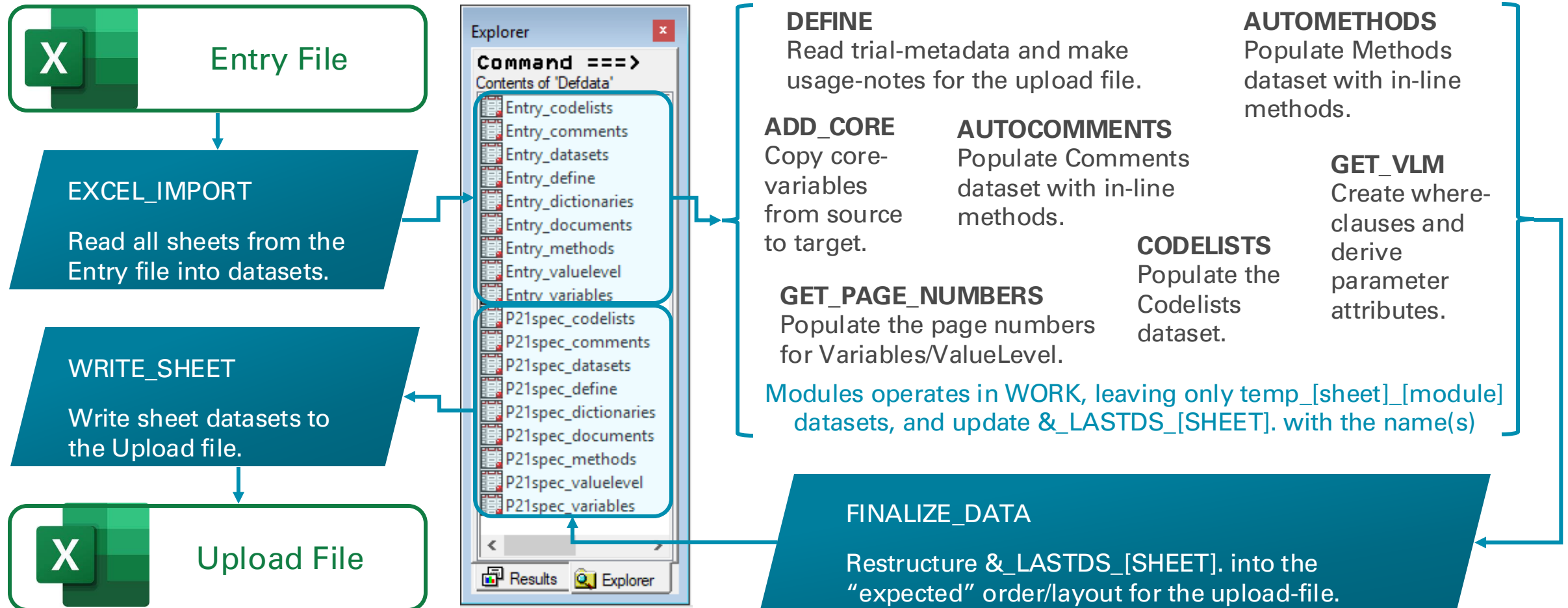
P21_upload.xlsx

Auto-generated and centralized metadata

Write-protected P21-compatible Define Excel

Support Datasets, Define.xml, aCRF creation & Rawdata-SDTM mappings

Modules – and their role in life



Core Variable Groups – automated copy/paste

1. In the **Entry file, Variables** sheet *Core Groups* are:
 - a. Defined by an integer in the Core Group column.
 - b. Referred to by a “code” in the Variable column.

	A	B	C	D	E	AA
1	Dataset	Variable	Label	Data Type	Length	Core Group
2	ADSL	STUDYID	Study Identifier	text	25	1
3	ADSL	USUBJID	Unique Subject Identifier	text	50	1
15	ADSL	ARM	Description of Planned Arm	text	50	2
16	ADSL	ACTARM	Description of Actual Arm	text	50	2
144	ADAE	:CORE.ADSL1				
145	ADAE	:CORE.ADSL2				
146	ADAE	AESEQ	Sequence Number	integer	8	
478	ADMK	:CORE.ADSL1				
479	ADMK	AVISIT	Analysis Visit	text	25	

This feature is not limited to ADAM.ADSL – though it is the most frequently used source.

	B	C	D	E	F	Q
1	Dataset	Variable	Label	Data Type	Length	Predecessor
2	ADSL	STUDYID	Study Identifier	text	25	
3	ADSL	USUBJID	Unique Subject Identifier	text	50	
15	ADSL	ARM	Description of Planned Arm	text	50	
16	ADSL	ACTARM	Description of Actual Arm	text	50	
118	ADAE	STUDYID	Study Identifier	text	25	ADSL.STUDYID
119	ADAE	USUBJID	Unique Subject Identifier	text	50	ADSL.USUBJID
122	ADAE	ARM	Description of Planned Arm	text	50	ADSL.ARM
123	ADAE	ACTARM	Description of Actual Arm	text	50	ADSL.ACTARM
124	ADAE	AESEQ	Sequence Number	integer	8	
744	ADMK	STUDYID	Study Identifier	text	25	ADSL.STUDYID
745	ADMK	USUBJID	Unique Subject Identifier	text	50	ADSL.USUBJID
746	ADMK	AVISIT	Analysis Visit	text	25	

2. In the **Upload file, Variables** sheet:
 - a. The *Core Group Variables* are copied to the “targets”.
 - b. Set to be predecessor of the “source”.

Automethods – Specify Methods in-line or use existing standard methods.

	A	B	O	P	Q	R
	Dataset	Variable	Method	Method Expression Context	Method Expression Code	Method Document
1						
11	ADMO	ADT	Numeric date of image	SAS 9.4	input(IMGDT,yymmdd10.)	SAP
12	ADMO	ADY	GEN_MET.ADY			

	A	B	C	D	E	F
	ID	Name	Type	Description	Expression Context	Expression Code
1						
2	GEN_MET.ADY	General syntax for ADY	Computation	Analysis Relative Day	SAS 9.4	ADT - TRTSDT + (ADT ge TRTSDT)
3	ADMO_ADT_AUTOMET	Syntax for ADMO.ADT	Computation	Numeric date of image	SAS 9.4	input(IMGDT,yymmdd10.)

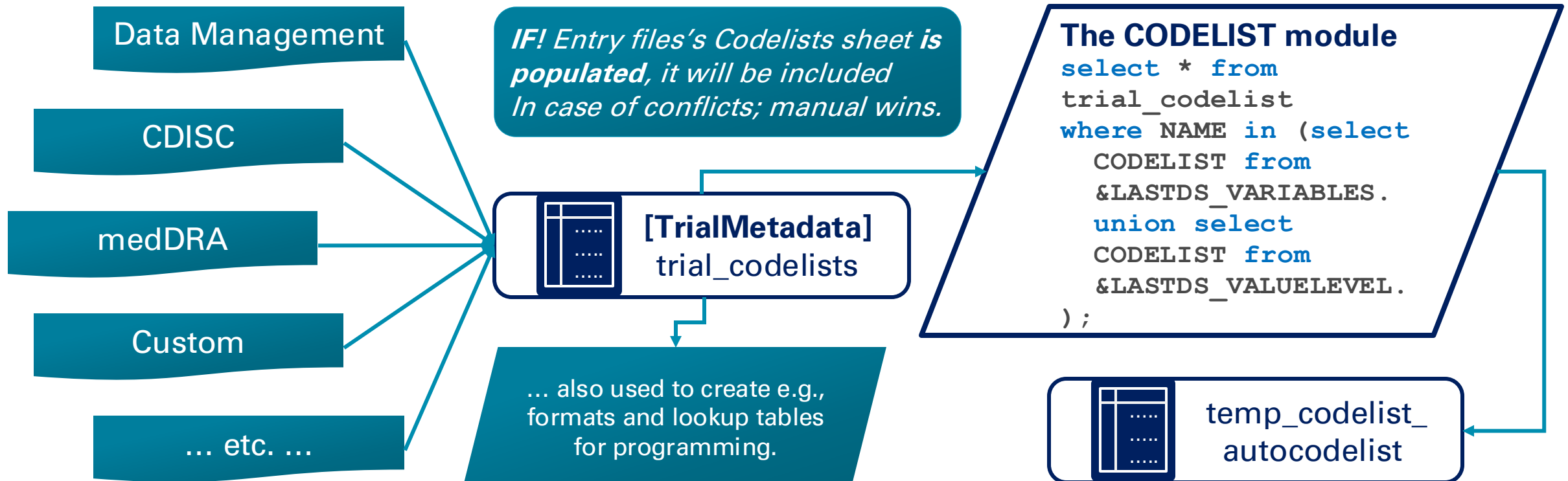
	A	B	C	P
	Order	Dataset	Variable	Method
1				
4	11	ADMO	ADT	ADMO_ADT_AUTOMET
5	12	ADMO	ADY	GEN_MET.ADY

Same general concept is used for Comments.

- In the **Entry file Variables (or Value Level)** sheet, specify methods as:
 - In-line methods:** by filing in Expression Context/Code and Document/pages.
 - Standard methods:** by specifying the ID of an existing method.
- In the **Upload file, Methods** sheet:
 - In-line methods:** Are populated with data from the Variables sheet, and ID/Name are created.
 - Standard methods** that are referred from the Variables sheet are left as-is.
- In the **Upload file, Variables** sheet:
 - In-line methods:** Are updated with the ID created in the Methods sheet.
 - Standard methods** are left as-is.

Codelists

- The **Codelists** sheet in the **Entry file** is – *generally* – not filled in.
- The **Codelists** sheet in the **Upload file** is automatically populated from a trial-level repository.



VLM Generation using GET_VLM module

Purpose

Automates VLM creation, reducing manual effort while allowing manual specification in entry file for exceptions.

Process Flow

Reads **VLM Target Variables** and **Selection Criteria** from entry specs, applies defaults when unspecified, and builds **WHERE clauses** while detecting value types, labels, and lengths directly from datasets.

Conflict Handling

Manual entries are the source of truth when domain/variables or attributes conflict. Conflicts are flagged with **VLM_SOURCE flag (AUTO/MANUAL)** for programmer's review and curation.

Auto-generated and manual VLM records are merged, marked with a VLM_SOURCE flag, and exported to a write-protected Excel sheet for programmer's discretion, traceability and conflict detection.

Beyond *Define.xml*

aCRF Generation

GET_PAGE_NUMBERS extracts CRF page numbers and variable names

Auto-generates annotations from EDC design specs to aCRF.PDF

Reduces manual effort, supports continuous updates, automates annotations

Raw Data -> SDTM Mappings

Compares **EDC Specs** with **Entry file** to check rawdata-to-SDTM mappings.

Flags missing mappings or typos — ensures completeness and alignment

Cannot check correctness but aids programmers review and update the SDTM specs efficiently.

Summary

Robust Framework

The new framework transforms how we manage clinical study metadata and generate submission deliverables—from initial specs to final submission.

User-Friendly Interface

By combining familiar P21 Excel templates with automated SAS tools, we have created a workflow that is not only efficient and reliable, but also easy for teams to use every day.

Modular design and Automation

The modular design means we can adapt and scale as our needs change, while automation reduces manual work and errors.

Enhanced Data Quality

Faster, compliant, and high-quality submissions, improved collaboration, and more time for innovation and skill development.

Thank you.

Malathi Hari
Senior Specialist, Statistical Programming
mhi@ascendispharma.com

ascendispharma.com

Thomas Gulholm-Hansen
Principal Consultant
tgu@tguconsult.dk



*Get it on
GitHub*