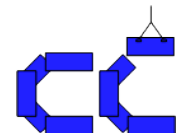


RESULTS-LEVEL METADATA: WHAT, HOW, AND WHY

Frank Dilorio, CodeCrafters, Inc.

Jeffrey Abolafia, Rho, Inc.



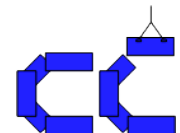
SOME BACKGROUND

CDISC: provide end-to-end standards

Already have “upstream” standards: CDASH, SDTM, ADaM
2015 release of the “other” end: Analysis Results Metadata (ARM)
This is an extension of the define-xml version 2 standard

CDISC now provides

End-to-end standards
Full traceability from data collection to results



SCOPE OF THIS PRESENTATION

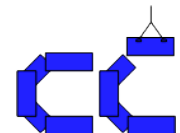
Discuss history of ARM

Make the business case for its use

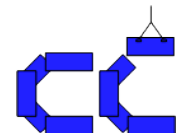
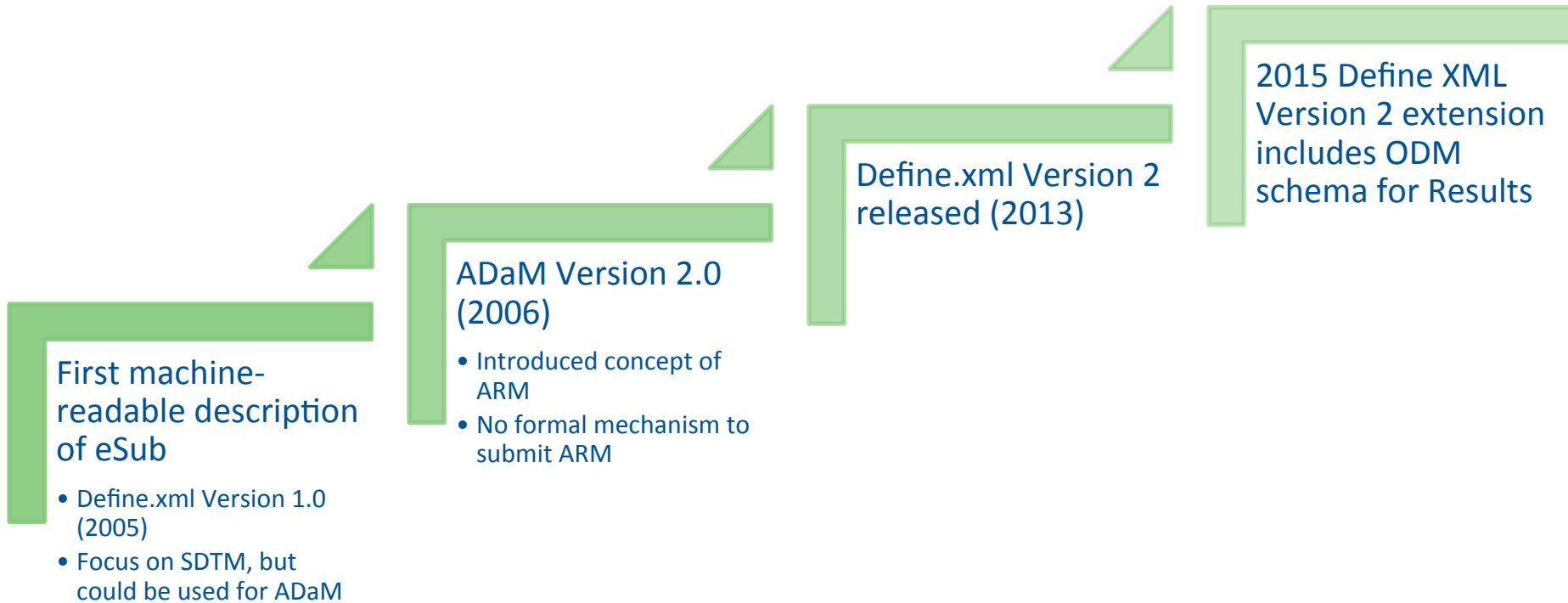
Outline implementation strategies

- Storage: Database
- Entry: Interface
- Coding: Creating the XML

Throughout: identify costs/benefits of different strategies



HISTORY OF ARM



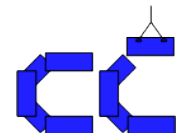
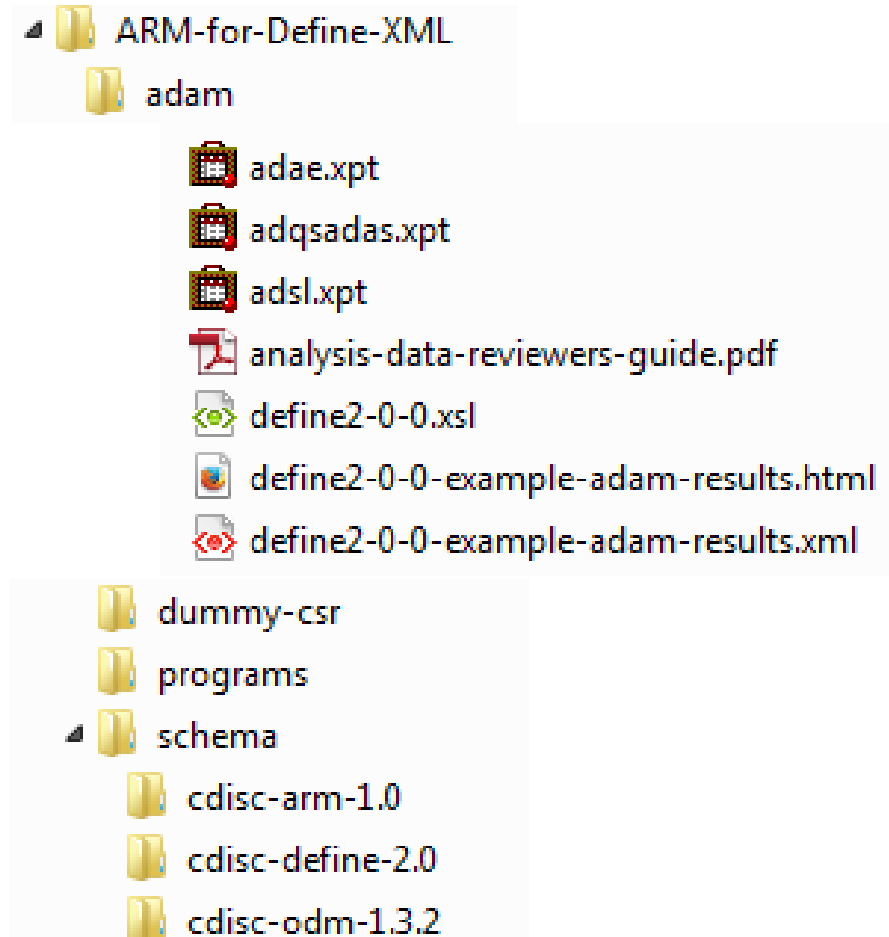
ARM RELEASE PACKAGE

Sounds good ... but
where to start?

- As always, best place is
the CDISC web site

www.cdisc.org/adam

- ZIP file with schema,
specification document,
and ADaM define.xml
examples

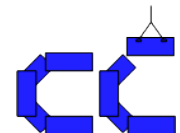


WHY BOTHER WITH ARM? ITS **NOT** REQUIRED

It must be costly to modify databases, interfaces and software to produce ARM...

Look at benefits:

- ✓ The documentation contained in ARM aids traceability
- ✓ This in turn can hasten time to regulatory approval
- ✓ Same metadata can be used throughout the TFL life cycle as with datasets/variable metadata
 - For creation of TFLs (header/footer, subsetting, etc.)
 - As part of project management (discussed later)



IMPLEMENTATION



ARM “tells the story” of a TFL; how do we collect, store, and present the pieces of the story?

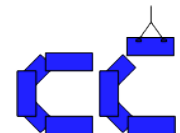


Next sections describe our ARM experience with:

- Database design
- User interface and metadata entry
- Creation of ARM-compliant define.xml



First, though, let’s read ahead and see the finished story ...



TRANSFORMED ARM (FROM CDISC RELEASE PACKAGE)

Style sheet transforms define.xml into HTML that presents each display in a readable, highly navigable/hyperlinked table.

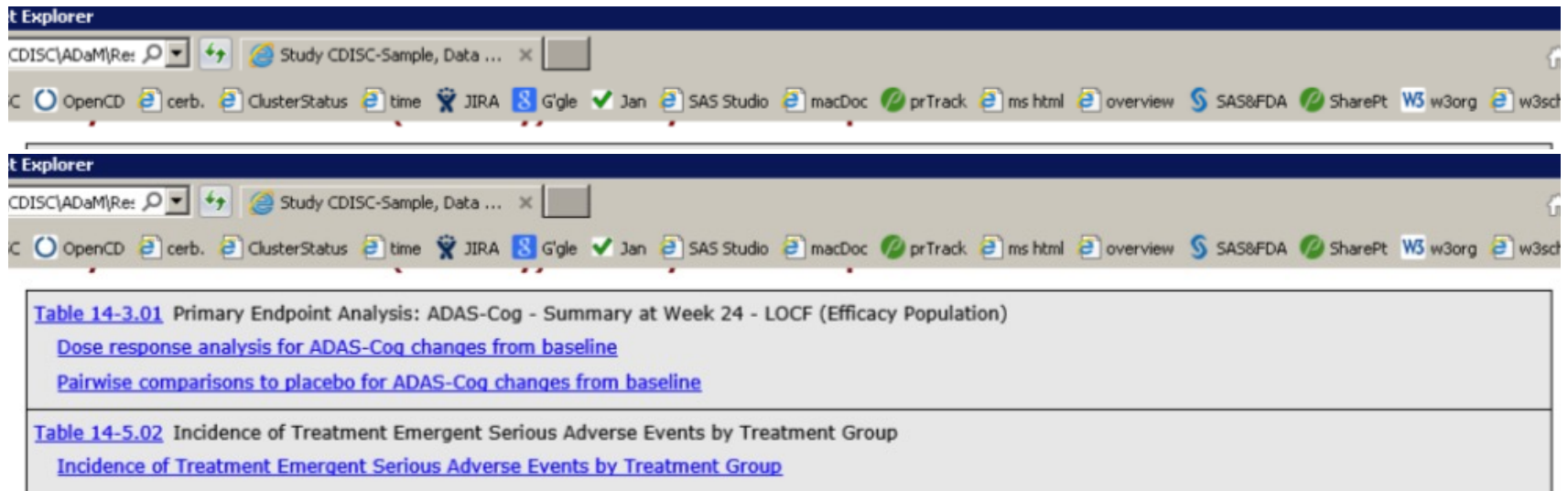
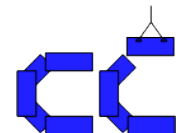


Table 14-3.01 Primary Endpoint Analysis: ADAS-Cog - Summary at Week 24 - LOCF (Efficacy Population)
Dose response analysis for ADAS-Cog changes from baseline
Pairwise comparisons to placebo for ADAS-Cog changes from baseline
Table 14-5.02 Incidence of Treatment Emergent Serious Adverse Events by Treatment Group
Incidence of Treatment Emergent Serious Adverse Events by Treatment Group

Analysis Results Metadata (Detail) for Study CDISC-Sample

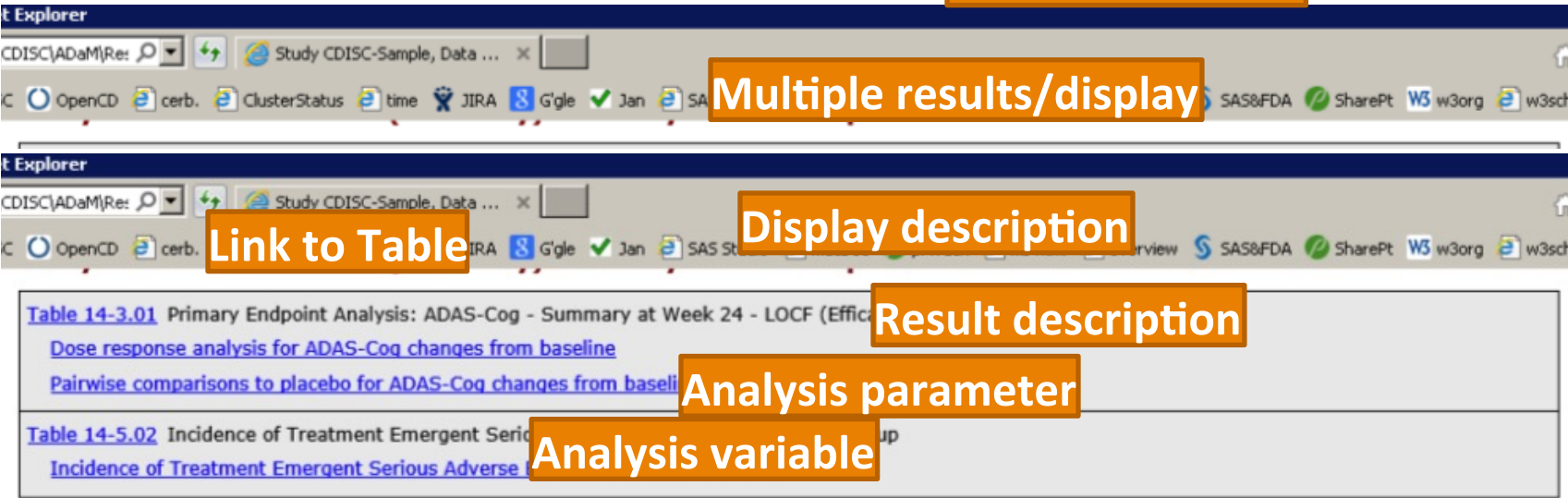
Table 14-3.01

Display	Table 14-3.01 Primary Endpoint Analysis: ADAS-Cog - Summary at Week 24 - LOCF (Efficacy Population)
Analysis Result	Dose response analysis for ADAS-Cog changes from baseline
Analysis Parameter (s)	PARAMCD = "ACTOT" (Adas-Cog(11) Subscore)
Analysis Variable (s)	CHG (Change from Baseline)



TRANSFORMED ARM (FROM CDISC RELEASE PACKAGE)

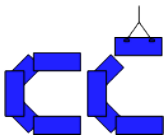
Style sheet transforms define.xml into HTML that presents each display in a readable, highly navigable/hyperlinked table. [Table of Contents](#)



Analysis Results Metadata (Detail) for Study [Dataset filtering](#)

Table 14-3.01

Display	Table 14-3.01 Primary Endpoint Analysis: ADAS-Cog - Summary at Week 24 - LOCF (Efficacy Population)
Analysis Result	Dose response analysis for ADAS-Cog changes from baseline Result documentation
Analysis Parameter (s)	PARAMCD = "ACTOT" (Adas-Cog(11) Subscore)
Analysis Variable (s)	CHG (Change from Baseline) Program snippet (can also include link to program)



TELLING THE STORY, PART 1: THE DATABASE

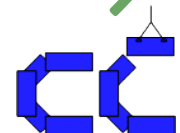


“Chatty” story means lots of metadata

Database design needs to be carefully thought out

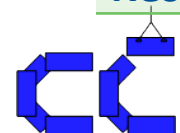
CDISC specification describes naming, content, status and cardinality of fields

Most of these are summarized in the next slide ...



ARM FIELDS

Level	Item	Required?
Display	Unique display name-number	Yes
Result	Title of the Display	Yes
Result	Analysis reason	Yes
Result	Analysis purpose	Yes
Result	Description	Yes
Result	Reference to external documentation	No
Result	Description of how to join multiple datasets used	Conditional
Result	Documentation (in-line text)	No
Result	Documentation (reference to external document)	No
Result	Programming code	No
Result	Software reference (e.g. "SAS Version 9.3")	No
Result	Link to the program that created the Result	No
Result-dataset	Name of dataset used for the Result	Yes
Result-dataset	Dataset filter (WHERE clause) for PARAMCD-based Result	Conditional
Result-dataset	Variables used to create the analysis result	Conditional



FIELDS AND REPETITION

- Note that just as ARM is not required for every display, there are also optional and conditional fields.
- “Level” implies hierarchy (cardinality is in **bold**):

Display description, documentation [1]

Result **[1 or more]**

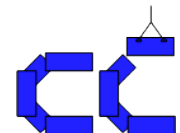
Description [1]

Datasets and variables **[1 or more]**

Documentation [1]

Programming code [1]

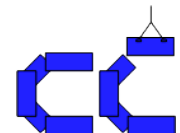
The database must be able to accommodate multiple results, datasets per display.



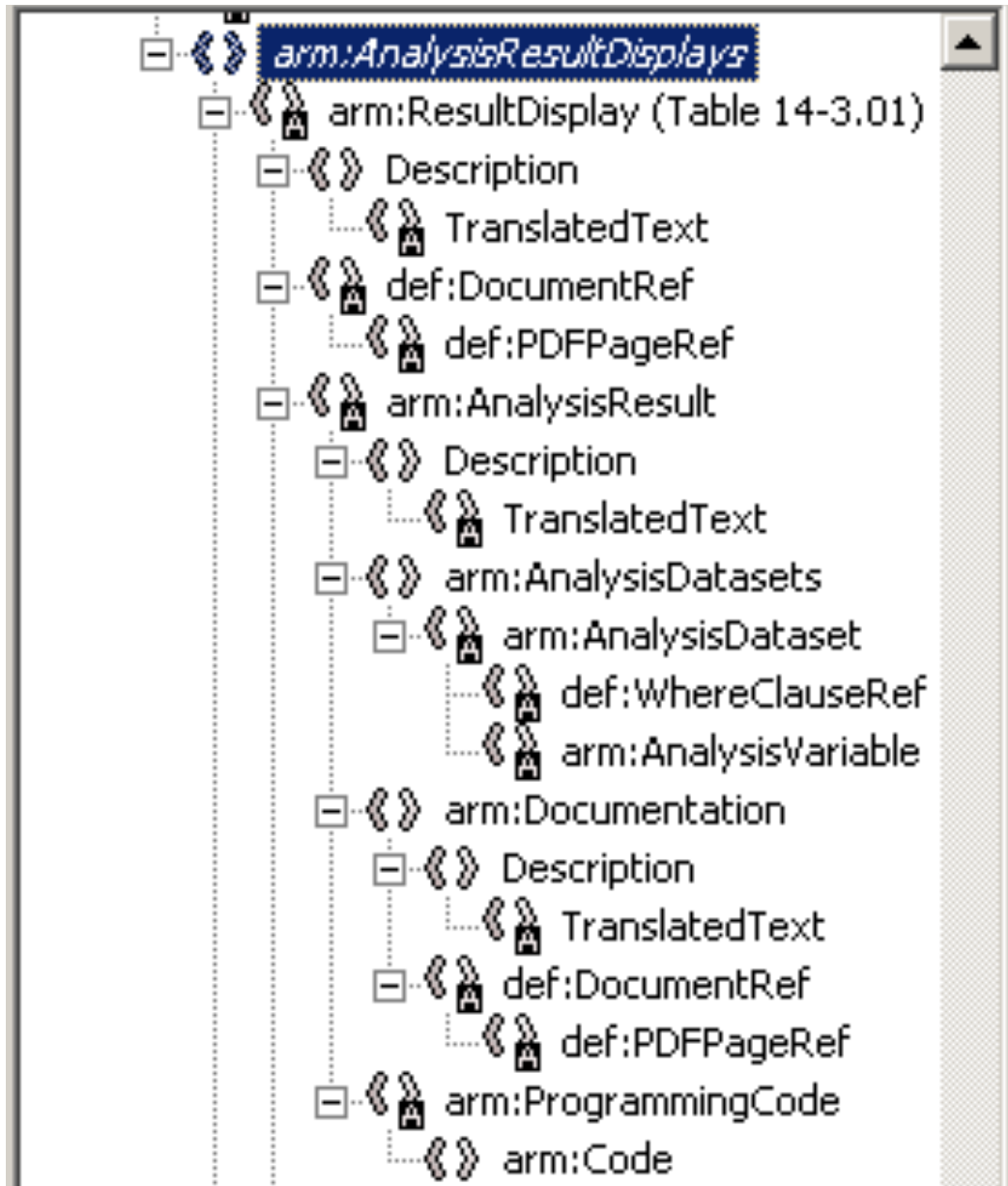
HIERARCHY ILLUSTRATED

Level of indentation corresponds with level of hierarchy:

```
<arm:ResultDisplay OID="RD.Table_14-5.02" Name="Table 14-5.02">
  <Description>
    <TranslatedText>Incidence of Serious AEs</TranslatedText>
  </Description>
  <def:DocumentRef leafID="LF.Table-14-5.02">
    <def:PDFPageRef PageRefs="3" Type="PhysicalRef"/>
  </def:DocumentRef>
  <arm:AnalysisResult OID="AR.Table_14-5.02.R.1"
    AnalysisReason="SPECIFIED IN SAP"
    AnalysisPurpose="PRIMARY OUTCOME MEASURE">
    <Description>
      <TranslatedText>Trt-Emergent Serious AEs</TranslatedText>
    </Description>
    <arm:AnalysisDatasets def:CommentOID="COM.JOIN-ADSL-ADAE">
      <arm:AnalysisDataset ItemGroupOID="IG.ADAE">
        <def:WhereClauseRef WhereClauseOID=
          "WC.Table_14-5.02.R.1.ADAE" />
        <arm:AnalysisVariable ItemOID="IT.ADAE.AEBODSYS"/>
      </arm:AnalysisDataset>
    </arm:AnalysisDatasets>
  </arm:AnalysisResult>
</arm:ResultDisplay>
```



HIERARCHY ILLUSTRATED USING XMLPAD



TELLING THE STORY, PART 2: POPULATING ARM

2 Approaches

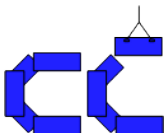
Interface: extend capabilities of existing system

- Can't just add new fields to existing "flat" structure. Have to add screens, links for metadata that repeats (results, datasets)

Programmatic: populate XLS/ other, and load into Oracle

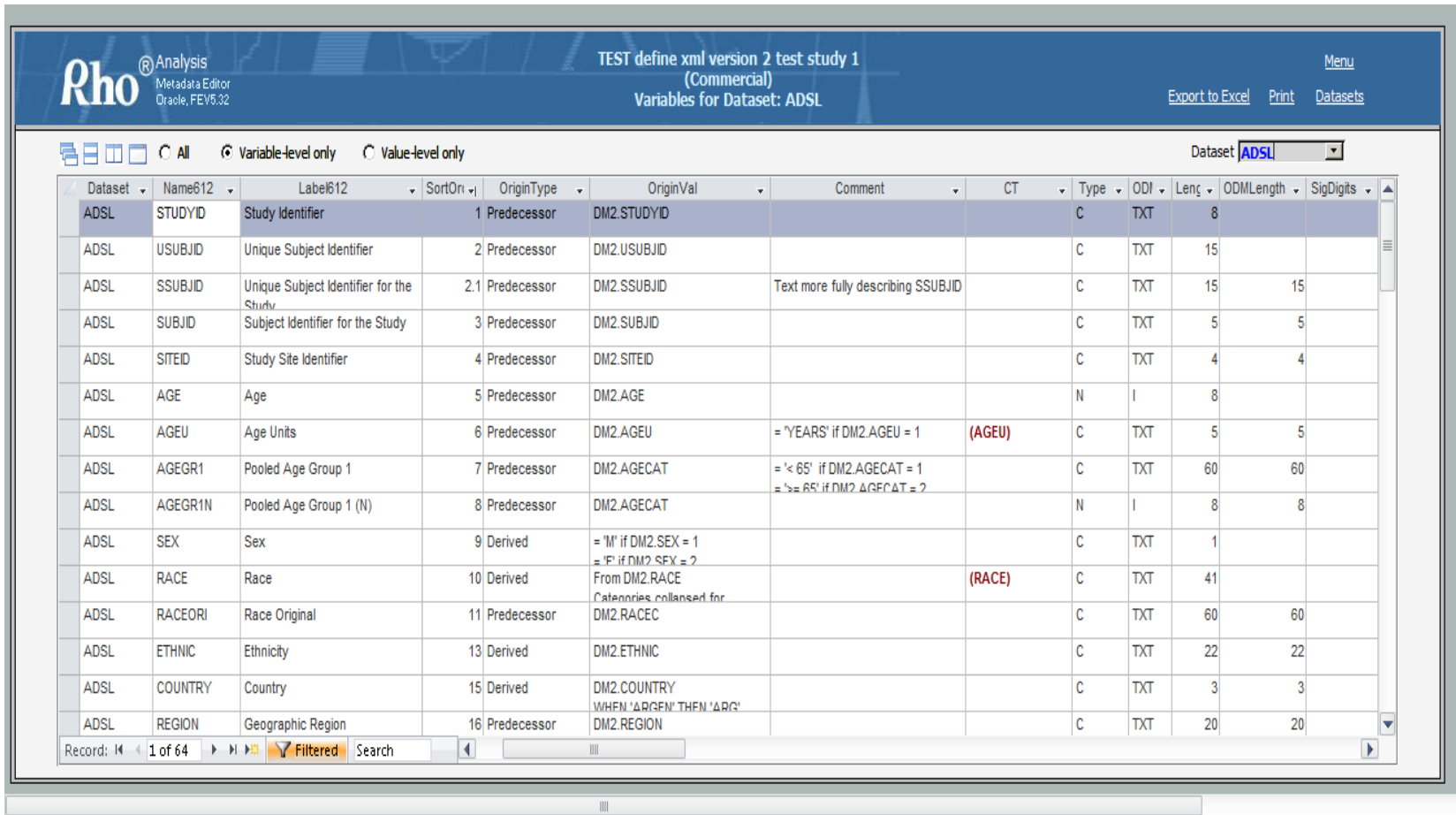
- Makes repetition of rows (e.g., results) easier than using the interface

Considering end-user needs, it's best to have both options available!



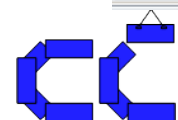
POPULATING ARM: SAMPLE INTERFACE

Look and feel for ARM resembles what we currently use for entry of dataset/variable/value metadata



The screenshot displays the Rho Analysis Metadata Editor interface. The title bar indicates 'TEST define xml version 2 test study 1 (Commercial)' and 'Variables for Dataset: ADSL'. The interface includes a menu bar with 'Menu', 'Export to Excel', 'Print', and 'Datasets'. Below the menu bar, there are tabs for 'All', 'Variable-level only', and 'Value-level only'. The main area shows a table of variables for the 'ADSL' dataset. The table has columns for Dataset, Name612, Label612, SortOrder, OriginType, OriginVal, Comment, CT, Type, ODI, Length, ODMLength, and SigDigits. The variables listed include STUDYID, USUBJID, SSUBJID, SUBJID, SITEID, AGE, AGEU, AGEGR1, AGEGR1N, SEX, RACE, RACEORI, ETHNIC, COUNTRY, and REGION. Some variables have comments and CT values, such as (AGEU) for AGEU and (RACE) for RACE. The bottom status bar shows 'Record: 1 of 64' and a search bar.

Dataset	Name612	Label612	SortOrder	OriginType	OriginVal	Comment	CT	Type	ODI	Length	ODMLength	SigDigits
ADSL	STUDYID	Study Identifier	1	Predecessor	DM2.STUDYID			C	TXT	8		
ADSL	USUBJID	Unique Subject Identifier	2	Predecessor	DM2.USUBJID			C	TXT	15		
ADSL	SSUBJID	Unique Subject Identifier for the Study	2.1	Predecessor	DM2.SSUBJID	Text more fully describing SSUBJID		C	TXT	15	15	
ADSL	SUBJID	Subject Identifier for the Study	3	Predecessor	DM2.SUBJID			C	TXT	5	5	
ADSL	SITEID	Study Site Identifier	4	Predecessor	DM2.SITEID			C	TXT	4	4	
ADSL	AGE	Age	5	Predecessor	DM2.AGE			N	I	8		
ADSL	AGEU	Age Units	6	Predecessor	DM2.AGEU	= 'YEARS' if DM2.AGEU = 1	(AGEU)	C	TXT	5	5	
ADSL	AGEGR1	Pooled Age Group 1	7	Predecessor	DM2.AGECAT	= '< 65' if DM2.AGECAT = 1		C	TXT	60	60	
ADSL	AGEGR1N	Pooled Age Group 1 (N)	8	Predecessor	DM2.AGECAT	= '>= 65' if DM2.AGECAT = 2		N	I	8	8	
ADSL	SEX	Sex	9	Derived	= 'M' if DM2.SEX = 1 = 'F' if DM2.SEX = 2			C	TXT	1		
ADSL	RACE	Race	10	Derived	From DM2.RACE Categories collapsed for		(RACE)	C	TXT	41		
ADSL	RACEORI	Race Original	11	Predecessor	DM2.RACEC			C	TXT	60	60	
ADSL	ETHNIC	Ethnicity	13	Derived	DM2.ETHNIC			C	TXT	22	22	
ADSL	COUNTRY	Country	15	Derived	DM2.COUNTRY WHEN 'ARGEN' THEN 'ARG'			C	TXT	3	3	
ADSL	REGION	Geographic Region	16	Predecessor	DM2.REGION			C	TXT	20	20	



POPULATING ARM: SAMPLE INTERFACE

Hide/Show Columns

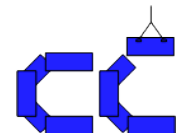
CSV

PDF

Print

Search:

Name	Links	Status	Title	Population	Group	Number	Validation Method
TAD_DM_01		Started	Demographic and Baseline Characteristics	Intent-to-Treat	Demographics	14.1.3	Double Programming
TAD_DM_02		Started	Demographic and Baseline Characteristics	Clinically Evaluable	Demographics	14.1.4	Double Programming
TAD_DM_03		Started	Demographic and Baseline Characteristics	Enrolled Safety	Demographics	14.1.5	Double Programming
TBI_EX_01		Started	Exposure to Final Dose by Final Dose Group	Enrolled Safety	Exposure and Compliance	14.3.5.5	Double Programming
TBI_EX_02		Started	Exposure to Final Dose by Final Dose Group	Randomized Safety	Exposure and Compliance	14.3.5.6	Double Programming
TAG_EX_01		Started	Exposure to Treatment	Enrolled Safety	Exposure and Compliance	14.3.5.1	Double Programming
TAG_EX_02		Started	Exposure to Treatment	Randomized Safety	Exposure and Compliance	14.3.5.2	Double Programming
TAH_FD_01		Started	Final Dose by Subgroups	Enrolled Safety	Exposure and Compliance	14.3.5.3	Double Programming
TAH_FD_02		Started	Final Dose by Subgroups	Randomized Safety	Exposure and Compliance	14.3.5.4	Double Programming
TAX_AEO_01		Started	Overview of Adverse Events During Double-Blind Phase	Randomized Safety	Adverse Events	14.3.1.3	Double Programming
TAW_AEO_03		Started	Overview of Adverse Events During Entire Study	Enrolled Safety	Adverse Events	14.3.1.4	Double Programming
TAW_AEO_01		Started	Overview of Adverse Events During Open-Label Phase	Enrolled Safety	Adverse Events	14.3.1.1	Double Programming
TAW_AEO_02		Started	Overview of Adverse Events During Open-Label Phase	Randomized Safety	Adverse Events	14.3.1.2	Double Programming
TBE_VS_02		Started	Potentially Clinically Significant (PCS) Vital Sign Values During Double-Blind Phase	Randomized Safety	Vital Signs	14.3.5.10	Double Programming
TBE_VS_01		Started	Potentially Clinically Significant (PCS) Vital Sign Values During Entire Study	Enrolled Safety	Vital Signs	14.3.5.9	Double Programming

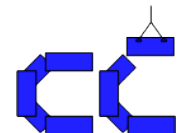


TELLING THE STORY, PART 3: CREATING THE XML

Results XML is defined by an ODM schema extension (in CDISC release package)

Transformation into HTML is handled by style sheet/XSL (also in release package) that is ARM aware

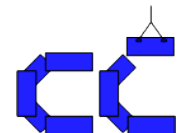
Building Results XML is no different than building earlier versions of define.xml



TELLING THE STORY, PART 3: CREATING THE XML (CONT.)

Regardless of software used to create the XML, keep in mind:

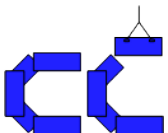
- OID (**O**bject **I**Dentifier) formation is critical
 - define.xsl creates links from results section back to datasets & variables. Therefore, naming has to be consistent.
- Exploit repetition
 - Results XML has elements (e.g., Description, WhereClause) found elsewhere in define.xml. Turn creation of these elements into a macro/function.



OID FORMATION

Highlighted OIDs are part of non-Results XML. Naming must be identical here in Results in order for style sheet to create link successfully.

```
<arm:ResultDisplay OID="RD.Table_14-5.02" Name="Table 14-5.02">
  <Description>
    <TranslatedText>Incidence of Serious AEs</TranslatedText>
  </Description>
  <def:DocumentRef leafID="LF.Table-14-5.02">
    <def:PDFPageRef PageRefs="3" Type="PhysicalRef"/>
  </def:DocumentRef>
  <arm:AnalysisResult OID="AR.Table_14-5.02.R.1"
    AnalysisReason="SPECIFIED IN SAP"
    AnalysisPurpose="PRIMARY OUTCOME MEASURE">
    <Description>
      <TranslatedText>Trt-Emergent Serious AEs</TranslatedText>
    </Description>
    <arm:AnalysisDatasets def:CommentOID="COM.JADSA100001.AE">
      <arm:AnalysisDataset ItemGroupOID="IG.ADAE">
        <def:WhereClauseRef WhereClauseOID=
          "WC.Table_14-5.02.R.1.ADAE" />
        <arm:AnalysisVariable ItemOID="IT.ADAE.AEBODSYS"/>
      </arm:AnalysisDataset>
    </arm:AnalysisDatasets>
  </arm:AnalysisResult>
</arm:ResultDisplay>
```



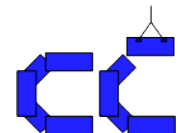
MAKING XML VIEWABLE: THE STYLE SHEET

- XML is plain text and virtually unreadable.
- XSL (style sheets) transforms XML into HTML when the XML is opened in Windows Explorer or similar.

Keys points:

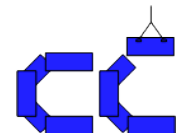


- XSL is conceptually different than other languages (so, generally, a badly shaped learning curve)
- Use the CDISC style sheet (in release package) unless it doesn't *fully* meet your needs.



IS THE XML CORRECT? VALIDATION

- Two levels of validation:
 - Syntax: the easy one (errors displayed in browser if elements are not closed, attributes aren't quoted correctly, etc.)
 - Content (“semantic”): the harder one. Validation answers the question “is the content of the XML consistent with the ODM schema?”
 - Are elements in the correct order?
 - Are required elements and attributes present?
 - Do attributes have expected values?
- We have tools for this, right ...?



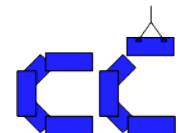
OPENCDISC VALIDATION

OpenCDISC has become the *de facto* standard for dataset and define.xml validation

Current (2015Q3) status of define.xml results validation:

- Not in Community (free, open source) version
- Will be in Enterprise (fee-based) version

Community version users will need to write their own checks. These can be created within the OpenCDISC validation framework.



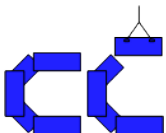
OTHER USES OF ARM

ARM is not just for “downstream”, eSub use

SAS macros or similar tools can use the metadata earlier in the study life cycle:

- Extract metadata to facilitate TFL production
- Create TFL template programs
- With additional fields in the database, can create macro variables containing TFL footnotes and other code fragments.

Management/tracking



FINAL THOUGHTS

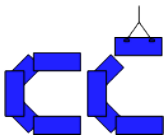
Good for regulatory reviewers and sponsors

Can speed time to drug approval.

Creation of database, means of ARM entry, and XML is not trivial.

Thoughtful planning and design, especially in concert with existing tools, can facilitate addition of this valuable metadata.

Building the infrastructure requires an up front investment but worth the investment!!



THANK YOU!

- Your comments and questions are appreciated and valued
- Contact the authors
 - Frank Dilorio: frank_dilorio@rhoworld.com
 - Jeff Abolafia: jeff_abolafia@rhoworld.com

