

The Next Logical Step Meta-programming

Greg Steffens

Noumena Solutions

noumena.solutions@gmail.com

February 23, 2017

PhUSE SDE Foster City, CA

Metaprogramming and Objectives

- Metaprogramming is highly reusable code that can be used across studies, TAs and companies
- It is necessary to evolve to metaprogramming in order to meet the objectives of conducting clinical trials
- We've done work to create data standards and to put those standards in metadata. The next logical step is to write code that reads the metadata-resident standards to automate the creation of standard data.
- Metaprogramming is the reason for metadata

Objectives – what they are

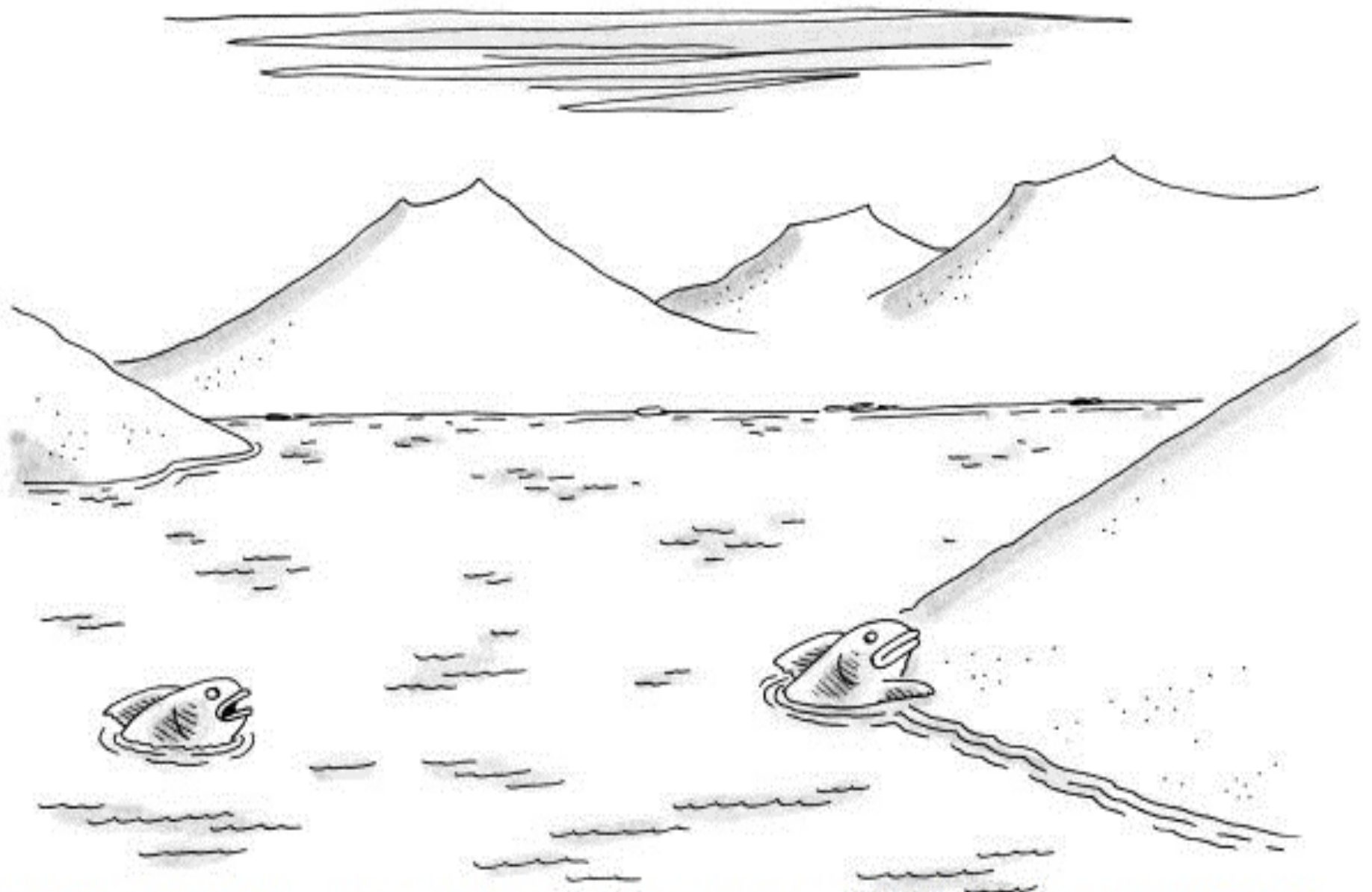
- To turn collected data into **information** for decision makers to be able to make the correct decisions about the safety and efficacy of treatments
 - To do this **quickly**
 - To do this in a highly **transparent** way, so that reviewers can quickly understand how the information was created
 - To do this with a consistent and high level of **quality**
 - To perform **analyses** of study information and **meta-analyses** of integrated data from multiple studies

Objectives – why they are important

- It is essential to define objectives, related to standards
- Without objectives, you can't measure your success nor evaluate the process and technology options
- When we clarify the objectives we see that standards are not the real objective, they are means to an end
- Lack of understanding of objectives has led to large projects and programs that resulted in ... not much, but with a lot of activity and noise

How are we doing so far?

- Very slow progress
- It still takes a long time to create even SDTM and ADaM, which is a distance away from being information
- Quality is inconsistent
- The process and technology is opaque, not transparent ... hidden mostly in SAS program files
- Data integration is not a central part of the study lifecycle and is done as needed
- We need to evolve, or else!



Why don't you just stay and work on being a better fish?

Periods in the Evolution of Standards



Where are you in the following evolution? Where is the industry? Are you a fish, amphibian, rodent, mammal or human? (Periods contain epochs contain ages.)

Period1: Inconsistency

- Epoch 1 of no data standards made it difficult to...
 - Pool data and do meta-analyses – the prime directive of CDISC and the FDA
 - Share program code across studies to get high quality with low effort
 - Be transparent about how data is collected, transformed, derived and reported
 - “Dinosaur managers” claimed they needed scientific freedom to design data differently, standards were deemed not possible or constraining
- Epoch 2 of Data Standards resident in documents and even early metadata (often in inadequate products like excel) did not adequately solve the problems. Compliance to standards and study specifications was difficult to implement and monitor. Lots of reentry of document-resident information into SAS programs. Sharing code was copy and tweak. Standards on a shelf.

Period2: Corporate Consistencies

- Epoch 1 of robust and rigorously standardized “**database metadata**” (DBMD). Realization that standardizing metadata is just as important as standardizing data because it supports metaprogramming.
- Epoch 2 of “**database metaprogramming**” (DBMP) started showing the real value and started to solve the problems. Moved beyond copy/tweak to macro libraries shared across all studies.
 - Specs in metadata does not do any more than Spec in documents without metaprogramming
 - Metadata: a standard list of database attributes put in a standardized database structure to support metaprogramming. Metaprogramming blurs the line between data and code – populating metadata does what writing programming code used to do.
 - But this was done in each company without adequate standards for the industry or even across TAs within a company. Standards? Yes, we have many!

Period3: Industry-Level Consistency

- So every company had standards – this didn't help the FDA do data pooling across companies.
- Epoch 1 of Industry-Level Standardization of data and DBMD
 - CDISC defines data standards in SDTM and ADaM. But can you easily pool cross-company data yet, whoops we need CT for the findings domain xxTESTCD
 - CDISC defines database metadata standards
 - Why aren't standards published in standard metadata?
 - Each CDISC team had their own variation of metadata that did not contain all the required attributes, instead of following the CDISC metadata standard (define.xml)
- Epoch 2 of Industry-Level shared software modules and industry DBMD
 - PROC REPORT is an example
 - We all use PROC REPORT but why not SAS macros for data transformations, deidentification, define.xml, etc. etc.?

Period4: Application of the Principles to Data Flow

- Epoch 1 of defining “**Map Metadata**” (MAPMD)
 - Map Metadata is standardized in structure
 - Map Metadata is standardized in content
- Epoch 2 where “**Map Metaprogramming**” (MAPMP) automates data flow
 - a Data Transformation Engine (DTE) that creates target databases from source databases based on simple mappings defined in map metadata.
 - Source DBMD, target DBMD, and MAPMD between them enable metaprogramming of data flow. Data flows from raw to SDTM to ADaM to IDB to reports and metadata has a parallel flow.

Where are We?

- Find where we are in the evolution and follow the steps to the next periods and epochs
- We need to reduce the amount of programming code that needs to be created for each study, to meet the objective of providing quality information quickly and transparently
- Creating new code takes time and increases the risk of errors
- Standards help because we can copy/tweak more code from study to study ... but only a little
- Maximize reusability of code across studies and processes

The myth of copy/tweak

- Code copied from study to study is often misconstrued as simple, high-quality, understood and transparent
- But the truth is that the code is changed by many people into complex “spaghetti” and very difficult to understand
- Study programmers learn the X number of tweaks to be made to the code without understanding the code and how it works
- Standards and copy/tweak will never get us to our objective

Metaprogramming Attributes

- Very highly reusable, rather than focused on one study or data flow
- Focused on underlying patterns, rather than surface processes
 - Data flows are the same underlying pattern, whether creating SDTM, ADaM or 3-rd party data
- Multi-use validated, rather than single-use validated for each study
- Driven by metadata, parameters and data; rather than manual tweaks
- Tends to use more advanced programming designs and techniques

Meta-X

- Metadata is data that describes data
- Metaprogramming is programming that generates program code
- It looks deeper under processes and problems to underlying patterns
- It uses coding techniques not used in study programming
- It looks complicated, in comparison to a study program but it isn't complicated – complicated just means “something I haven't seen yet”

The Flying Cars Retort

- I talked about the need for industry-standard metadata, row-level metadata, the data transformation engine that automates any data flow described in map metadata for years
- The reply from some was that I was talking about flying cars, meaning something like it was not possible
- It is not only possible but I've done it and proven it is actual use
- Metaprogramming is real and effective

News Announcement This Week

- A flying, autonomous drone that can carry people will begin "regular operations" in Dubai from July, the head of the city's Roads and Transportation Agency has announced at the World Government Summit.

Roles

- We all use things every day that we don't understand the inner workings of, but understand that it works
 - Proc report
 - iPhone
 - Cars
 - Pressing a key on your computer's keyboard
- A division of labor makes the world work
- As long as we all know that it works, not everyone needs to know how it works
- The role of a study programmer and that of a metaprogrammer are very different from each other

The DTE Metaprogram Examples

- Create an SDTM – compliant database with code like this

```
%dte(source_mdlib=m,source_prefix=raw_,target_mdlib=m,target_prefix=target_,maplib=me,inlib=raw,outlib=sdm,suppqual_make=yes)
```

- Publish your data requirements with code like this

```
%md2html(mdlib=m,inprefix=target_, maplib=m,mapprefix=raw_sdm_,html=yes,htmlfile=c:\metadata\target_define.html)
```

- Create the define.xml file with code like this

```
%md2odm(mdlib=m,outxml=c:\metadata\SDTM.xml,StudyName=SDTM_Standard 3.1.2,ProtocolName=SDTM,StandardName=SDTM 3.1.2,StudyDescription=SDTM Standard,defineVersion=1.0.0,odmversion=1.2,crt_prefix=def)
```

- Metadata contains the specification, metaprogramming publishes it and also creates the study data driven by it

Transparency Revisited

- Calling a metaprogram module like the DTE is transparent in several ways
- WHAT is done is defined in the DBMD and MAPMD
- How it is done is shown in the SAS log file with the MPRINT option turned on
- But 80% of that log file is processing metadata and implementing metalogic deciding what code needs to be generated to create a target
- The DTE can also generate a macro-free SAS program file that does everything required to create the target from the source

Familiar

- Metaprogramming, for us, is best done with SAS because that is the de facto industry standard for study programming
- SAS macros using structured programming design and relational metadata are familiar to study programmers

New Shiny Things

- There have been past attempts that have failed because of the solution being unfamiliar to study programmers. XML was initially to have been a replacement for SAS data and metadata, for example, but it was unfamiliar and there were no big advantages, so it didn't go as far as initially hoped
- The important thing is to understand what needs to be done, not what language to describe it in

Today

- We know where we are in the evolution and how far we need to go
- We see how metaprogramming applies to data and will see how that enables data visualization applications to present information to decision makers in a timely and quality manner
- We will see how someone outside our industry approaches technology solutions
- We will also see how to manage all the files in a part-11-compliant manner
- A panel discussion at the end of the day will allow a group discussion and more Q&A

