

Using Metadata to Integrate and Standardise Clinical Trial Data - the ICON Study Data Mapper

Paul Crean, ICON Clinical Research, Dublin, Ireland
Jennie McGuirk, ICON Clinical Research, Dublin, Ireland

ABSTRACT

To deliver clinical data for reporting, analysis and regulatory submission in a more efficient way, ICON has developed a data lifecycle and supporting toolset (the "Study Data Mapper") that uses metadata to standardise and conform clinical trial data. This standards-driven approach has improved the quality of study data production and achieved a considerable reduction in custom programming through the re-use of standard data models and transformations.

This paper will discuss the concepts behind the Study Data Mapper toolset and data lifecycle, the challenges that were encountered, and the lessons learned.

INTRODUCTION

Following on from a successful implementation of Oracle ® Life Sciences Data Hub in 2010, ICON embarked on a project to streamline the process of transforming and delivering clinical data.

The goals for this project were to reduce study setup times and increase ICON's flexibility to respond to customer requests through the standardisation and centralisation of data, reporting, and processes.

This resulted in the development of the ICONIK Clinical Data Lifecycle process and the ICON Study Data Mapper system.

The integration and standardisation of clinical data has reduced the duplication of data manipulation work and is increasing operational efficiency through standards-driven data processing and provision.

FUNDAMENTALS

The ICONIK Clinical Data Lifecycle involves the collection of raw clinical data from diverse systems and data structures; the integration and consolidation of clinical data into standard structures ("the hub"); and the distribution of conformed data to downstream data structures for use by data consumers. Target data consumers include Data Management for data cleaning, Biostatistics for SDTM and recruitment metrics for Clinical Operations.

The three pillars of the Clinical Data Lifecycle are:

- Standard Dataflow
- Centralised Processing
- Standard tools

STANDARD DATAFLOW

There is a diverse population of consumers of clinical data internally and externally to ICON. After reviewing the known requirements it became clear that a single physical data model, such as SDTM, could not meet all user requirements effectively. Because of this, the data lifecycle comprises multiple physical data models to provide flexibility and performance.

The Oracle Data Warehouse Reference Architecture was used as a guide in the development of the lifecycle, particularly the use of three conceptual layers for data staging, data foundation and data access (or data staging, data standardisation and data delivery in the ICONIK nomenclature).

The data models that have been implemented in each layer are optimized to reflect their intended purpose, for example data models in the data delivery layer are designed to reflect the requirements of their intended target audience and their preferred data presentation tool.

PhUSE 2012

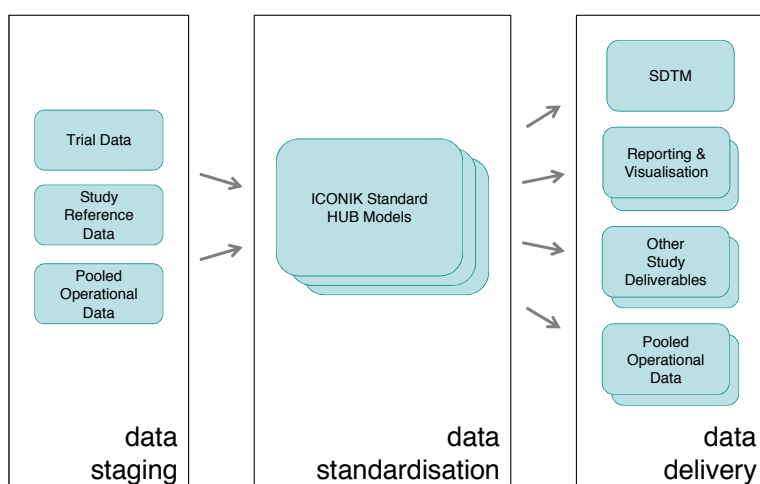


Figure 1: ICONIK Clinical Data Lifecycle

The lifecycle of all ICONIK clinical data models is maintained under version control using the Study Data Mapper system, which maintains central repositories of data structures, data maps, and study information.

CENTRALISED PROCESSING

A centralised Clinical Data Services team was created with responsibility for delivering clinical data services and reporting on new studies.

This team is responsible for clinical data standardisation and delivery. The team specifies and implements ETL programs to populate the ICON standard clinical data models; and generates reporting, SDTM and alternate sponsor deliverables.

The team is also responsible for continuing to develop and enhance the ICONIK clinical data models, programming libraries and standards tools and processes.

The team utilises the new tooling and standardised clinical data delivery processes that were developed by the project team.

STANDARD TOOLS

In addition to reengineering the processes for the transformation and delivery of clinical data, the tools that were used for these processes were also considered.

Transformation and delivery of data at ICON was very much a manual process with a high level of manual effort required to specify, program, maintain and deliver clinical data deliverables. Traditionally, study ETL work was being custom written from scratch on each study, with little reusability. Consequently there was little consistency in programming methods and styles, leading to additional validation effort.

Additionally programming teams were using a mixture of SAS® clients running on multiple platforms.

Once the high level architecture and guiding principles of the streamlined data lifecycle were agreed, a project team was created to gather user requirements and to develop a business case for replacing the existing tools with a standard toolset dedicated to data transformation. This group included representatives from Biostatistics & Programming, Data Management, Data Integration & Standardisation, and IT.

Based on the final user requirements ICON decided to develop a software application in-house. The Study Data Mapper is designed to manage the data structures used for conducting clinical research and for managing transformations between data structures in all three conceptual layers.

The Study Data Mapper also includes an optional interface to ICON's Oracle Life Sciences Data Hub (LSH) instance which hosts the Clinical Data Repository (ICON's central, global repository for clinical data).

ICON STUDY DATA MAPPER

The Study Data Mapper system is designed to:

1. Manage data model metadata including table, variable and value list related metadata;
2. Capture and manage transformation metadata; and
3. Generate standalone SAS programming code based on the recorded metadata.

PhUSE 2012

CORE FUNCTIONALITY

The Study Data Mapper consists of:

- An Oracle database hosting a central repository of metadata that is maintained under version control, including:
 - Metadata about study source data structures, ICON and sponsor standard target structures, and other supporting data.
 - Transformation metadata describing the mapping logic between target and source tables, variables and value lists.
 - Study metadata related to study characteristics, including sponsor and therapeutic area.
- An integrated Excel template to capture the metadata and to record the transformation logic (also referred to as “mapping”) between study source data and target data structures.
- A browser-based interface to allow the user to assemble all or part of a mapping project from the mapping components of previously stored projects in order to increase the overall efficiency of building mapping projects.
- A central library of pre-defined mapping functions that can be applied to one or more variables. A registry of functions is maintained within the metadata repository to support the extension of this library over time.
- Support for completeness/consistency checks for a mapping project that includes checks for incomplete or inconsistent mappings.
- A code generation engine that automatically generates the ETL programs that transform the data into the standard structures based on the captured mapping metadata. These generated ETL programs are stored in metadata under version control.
- A secured audit trail that is computer generated, time-stamped, and that independently records changes to the mapping projects and their component elements.

MAPPING PROCESS

The Study Data Mapper supports an iterative transformation specification process, in which study mapping can begin early in the process, information can be added as it becomes available, and the table mapping definitions can be finalized as they are completed.

The mapping process using the Study Data Mapper involves:

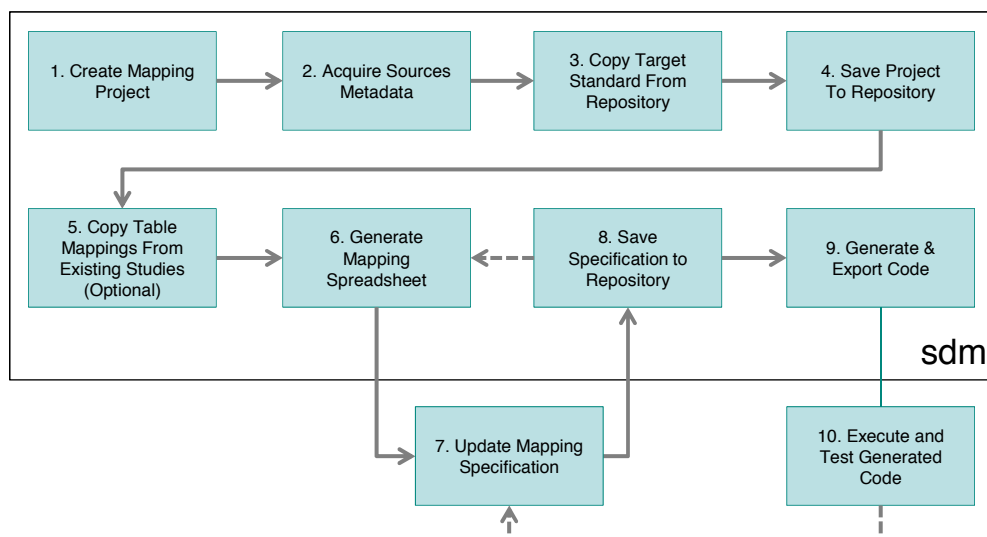


Figure 2: Mapping Process

1. **Create Mapping Project:** The mapping author creates a new project through the web interface, capturing mapping project related metadata (e.g., sponsor, study, therapeutic area). Multiple projects can be set up per study to support, for example “source-to-hub” and “hub-to-reporting”.
2. **Acquire Sources Metadata:** The mapping author imports the source data model metadata including metadata describing tables, variables, source system, and value lists. This metadata is imported from either an Excel file based on the SDM template or an SDM.XML file.

PhUSE 2012

3. **Copy Target Standard from Repository:** The mapping author identifies which standard data model is to be used as a target, and copies the target data model metadata into the project. Alternatively the target can also be loaded from an external file.
4. **Save Project to Repository:** The project is saved to the repository and the initial version of the mapping project is created, along with all of its components.
5. **Copy Table Mappings from Existing Studies (Optional):** The mapping author may optionally reuse some of the transformation logic from an existing project.

The mapping author selects the table maps that they want to copy from the original mapping project and they are presented with a wizard like multistep interface for the paste operation.

This wizard lists matches and non-matches at the table, variable, and value list level. For non-matches the mapping author can select an equivalent source table. For example, if both studies contain enrollment tables that are near identical in structure but differ in table name (ENRL and ENROLL), the mapping author can replace ENRL with ENROLL and the change will cascade to all maps that use ENRL as source. The mapping author may also choose to make the change local to the variable map, and prevent it from cascading to other variable maps.

This re-mapping functionality is also available for variables.

The user would like to utilize (or accept) the default values for the remaining steps they can finish the operation at any point.

6. **Generate Mapping Spreadsheet:** A mapping spreadsheet is generated within the web interface containing all metadata (data model and transformation) captured to date and is stored under version control in the SDM repository.
7. **Update Mapping Specification:** The mapping spreadsheet is the primary interface for capturing transformation metadata.

Within the mapping spreadsheet the user maintains metadata related to table map attributes (such as SQL JOIN clause), variable map attributes (such as mapping function, user-defined macro, source variables, parameters, and literals), and code list mapping attributes.

Data model metadata can also be maintained through the spreadsheet, including support for addition and modification of table and variable metadata. Source and target tables and variables may also be marked as "excluded" from the current mapping project.

Variables are mapped through a mapping operator. Mapping operators range from simple one-to-one assignments (DM.USUBJID= DEM.PT) to more complex operators involving one or more sources. All SAS functions that are compatible with PROC SQL are supported. Also a global library of validated mapping operators is available for common mapping functions, such as conversion of SAS numeric dates to ISO format.

If a suitable function is not available, the mapping author defines a custom operation within the specification providing details of the function name, expected parameters and parameter descriptions, and programmer notes. These details are used by the study programmer to develop the custom function.

8. **Save Specification to Repository:** At any point in the specification process, the mapping author can save the specification to the repository.

When the mapping author saves a spreadsheet to the Study Data Mapper repository, it is loaded into a staging area and its structure is checked for correctness. If the spreadsheet structure is correct, the user saves the spreadsheet to the repository.

As part of the save process the system uses a versioning mechanism to maintain and track multiple versions of mapping project entities (table, variable, table map, variable map, etc.) and their state. This provides the ability to reconstruct the state of the metadata for any given version

The mapping author may return to the specification process, as necessary, to supplement or revise the mapping specification.

9. **Generate & Export Code:** After a mapping author has saved the specification to the repository, they can generate ETL code for the mapping project. The generated code is stored in the repository under version control. The mapping author can export the current or any earlier version of the code for execution.

As part of the code generation, the system provides a completeness/consistency check for a mapping project that includes checks for variables that are unmapped but not marked as excluded, missing required variables, incomplete code list mappings and data type conversion issues.

This can be an iterative process as the mapping author continues to refine the table mappings over time.

PhUSE 2012

10. **Execute and Test Generated Code:** The exported code is imported into LSH and is updated to reference the global macro catalog containing the global library of mapping operators. If the mapping author has specified a custom mapping operation a reference to the study-specific macro catalog containing the custom coded SAS macro is also required.

On execution of the code, the mapping author reviews the outputs and decides whether to continue with the mapping process in SDM or to submit the specification and program code for review and testing.

The code follows a formal SDLC in LSH before it is submitted for automatic execution on a scheduled basis.

Alternatively, the generated code can be run standalone outside of LSH using compiled copies of the global macro catalog and the study macro catalog.

STANDARDS MANAGEMENT

A critical factor in achieving the ICONIK Clinical Data Lifecycle goal of supporting a move to the standardisation of data processing and data access is the implementation of code and data model libraries.

PROGRAMMING CODE OBJECTS

Programming code objects are developed and published within LSH. Programs, packages and macros must follow the LSH SDLC before they are made available for study use.

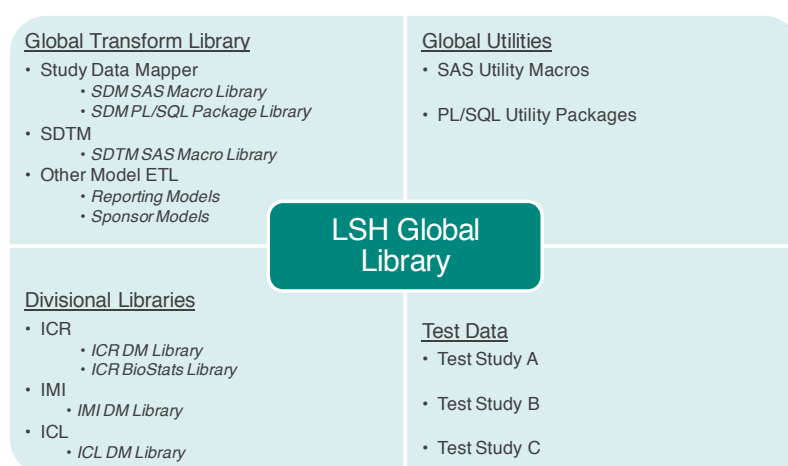


Figure 3: Code Library

Global macros and packages are available for use by all divisions and departments. Individual departments are responsible for the management of their department's library.

DATA OBJECTS

The Study Data Mapper metadata repository is used for the standards development lifecycle of the standard data models. The captured metadata includes version attributes (author, approver, version number, validation status, etc.) and search tags (e.g., therapeutic area, sponsor, etc.).

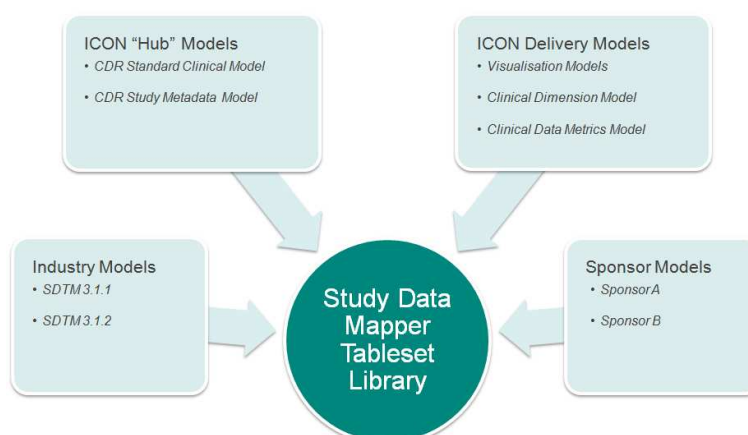


Figure 4: Data Objects Library

PhUSE 2012

CHALLENGES

The challenge was to develop a pragmatic solution that initially could be rolled out quickly with the capability to evolve over time. The project was very focused and made a purposeful decision not to embark on a multi-year journey to develop an ideal model or the ideal paper process.

The project team used the Scrum methodology throughout the project which encouraged collaboration and communication and helped the team focus on real-world requirements.

CONCLUSION

Experienced users of the Study Data Mapper can now map on average 90% of a study through the mapper without writing any custom code with an average of 8 custom macros being coded per study. The expectation is that these numbers will improve as the global library continues to grow and sponsor specific libraries are added.

On new studies, standard data for priority domains can be processed through the new lifecycle and be available shortly after EDC go live.

While many of the goals of this project have already been achieved, the process of improvement must be continued through the development of new capability.

REFERENCES

Enabling Pervasive BI Through a practical Data Warehouse Reference Architecture
<http://www.oracle.com/us/solutions/datawarehousing/058925.pdf>

RECOMMENDED READING

Agile/Evolutionary Data Modeling
<http://www.agiledata.org/essays/agileDataModeling.html>

CONTACT INFORMATION

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

Paul Crean
ICON Clinical Research
South County Business Park
Leopardstown, Dublin 18, Ireland
Work Phone: +353 (0) 1 291-2000
Fax: +353 (0) 1 291-2200
Email: paul.crean@iconplc.com
Web: www.iconplc.com

Jennie McGuirk
ICON Clinical Research (as above)
Work Phone: +353 (0) 1 291-2000
Fax: +353 (0) 1 291-2200
Email: jennie.mcguirk@iconplc.com
Web: www.iconplc.com

Brand and product names are trademarks of their respective companies.