Biomedical Concepts

A Treatise

Dave Iberson-Hurst
data4knowledge

Table of Contents

Document History

Version	Author(s)	Date	Changes
Draft 0.1	Dave Iberson-Hurst	2022-FEB-20	First draft for review. Contains the first four chapters and summaries of the content for the remaining chapters.
Draft 0.2	Dave Iberson-Hurst	2022-MAR-06	Addition of Chapter Five

Purpose of the Document

Over the last few years much has been heard about Biomedical Concepts (BCs). Unfortunately, this has not been followed by details such as designs, models, what they are or advantages of taking the BC road.

This paper is intended to answer those questions, present an initial design from which an “industry standard” can evolve, and discuss the advantages of using BCs via several use cases.

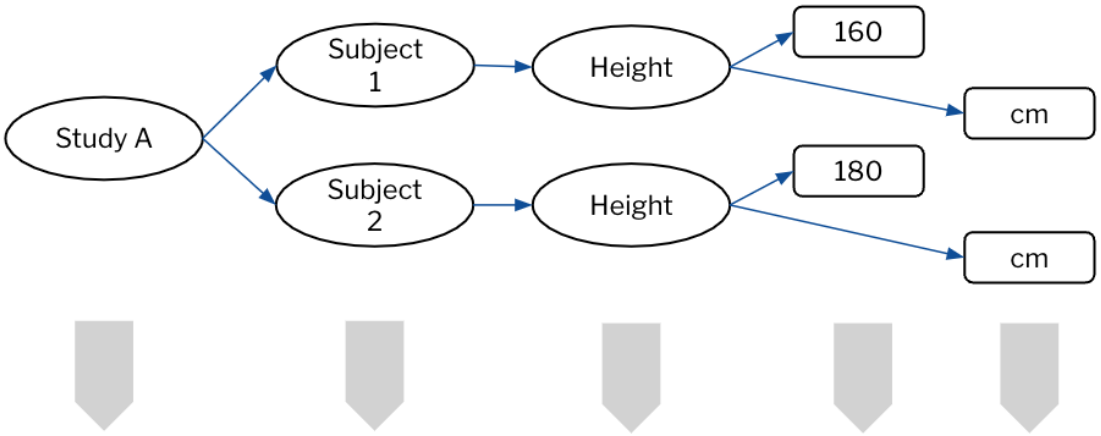
This paper will also make reference to the author’s practical experience of using BCs since first trying to implement BCs using, of course, MS Excel in 2012 and subsequent implementations using graph technologies.

Introduction

Relationships

It has been said many times, by several observers, that the current CDISC standards are views of our data rather than the actual data. The data we tabulate for example, is an extraction of that collected data. Define.xml is a view of the metadata of those tabulations, metadata being simply data. ODM is a view of the same data in a form structure, reflecting how it was collected. They are all but views of the data. The display of that collected data does not truly reflect the relationships within the data. For example we do not have explicit relationships between related columns in a tabulation, such as the result value and the result units but there is an obvious relationship there.

The simple example in Figure 1-1 shows an efficient way of drawing the data and how it is placed into a rectangular structure; the tabulation allows the human to consume but is not the best form to preserve the complexity of the relationships within the data, the relationships in the rectangular structure are implicit rather than explicit. It could be said that the data has a natural form.



Study	Subject	Test	Value	Units
A	1	Height	160	cm
A	2	Height	180	cm

Figure 1-1 - Natural or Tabular Form

We need several items of data to be joined to form a single coherent “observation” with those items being related in specific ways. The value without the units is of little use. Without an identifier for the “observation” we have useless data. We need a certain number of items to form something useful and we need the context in which the data were collected, the subject, the study and so forth.

This has been recognized for many years. The FDA published the first Study Data Technical Conformance Guide [Ref 1] back in February of 2014. In Appendix A, you will find these words

> *“A data value is by itself meaningless without additional information about the data (so called metadata). Metadata is often described as data about data. Metadata is structured information that describes, explains, or otherwise makes it easier to retrieve, use, or manage data. For example, the number 44 itself is meaningless without an association with Hematocrit. Hematocrit in this example is metadata that further describes the data. ”*

Those words remain in the conformance guide to this day.

Going forward, as the industry encounters more complex data, there will be a need to handle the relationships therein. Geospatial and address data contain many attributes and these are not rectangular!

The CDISC standards were developed in silos. This is not a criticism; it is just a fact of life. The problem space is a complex one and we, as humans, try to sub-divide that space so as to understand it. But by developing in silos, we have created silos in our data and we lose the relationships across the boundaries and this causes us difficulties when we wish to

automate [Ref 2]. We need a way to bridge those silos and, of course, the one item that bridges the silos is the data.

We want consistency in the data we capture so as to allow for the pooling of that data within sponsors and across industry. Currently this is hard to achieve. We should also recognize that our data can exist without the CDISC standards. I can measure my blood pressure so as to monitor my own health. A CDISC SDTM Vital Signs domain cannot exist without the necessary data. We want to be able to source the data for clinical research from multiple sources and allow for easy integration of such data. The data are independent from our human-enabled views of it. We need that independent data layer.

Machine Readable

As well as the main reasons for considering the move to Biomedical Concepts, there are a set of secondary reasons behind pushing for their implementation.

The first is the need for machine readable metadata. While we can load the standards into the machine, we are not getting the level of precision that would help with better quality data and checking. An example would be code list subsets for a specified test code.

Providing precision for each observation would allow Therapeutic Area Users Guides, the TAUGs, to be defined as a set of machine readable definitions with additional guidance documentation as to the use within studies.

There has always been the SDTM question of “where do I put X” and the proliferation of supplemental qualifiers. By bringing precision to the BC model CDISC could provide tighter guidance for sponsors, as to how to add supplemental qualifiers, when they are needed, while also allowing such additions to be readily recognized.

And all of these factors add to the ability to automate tasks across the study lifecycle.

Principles

To meet the needs outlined above BCs should be:

1. **Independent.** Each BC should be independent of the existing CDISC standards. A BC definition should not refer to any existing CDISC standard other than Controlled Terminology (CT).
2. **Linkable.** A BC should be able to be linked with the current CDISC standards but that link must be decoupled from the BC itself. We want links to existing standards for the purposes of automation but the BCs should be able to stand alone. Additionally, the BCs will be less likely to change than the views of the data such as SDTM and will allow for new standards to be developed more rapidly.
3. **Addressable.** Each observation should be uniquely identified and versioned such that I could use a single BC independently, each BC is addressable in its own right.
4. **Complete.** A BC definition should be complete, have all the necessary definitions such that it can be used directly, e.g. terminology references.

Summary

1. Relationships
    a. Move away from views of the data to the natural form of the data and complete relationships within the data
    b. Silos and missing relationships
    c. The data and the current CDISC standards are independent
    d. Quality and consistency
2. Machine Readable
    a. Therapeutic Areas
    b. Help industry with better “mapping” guidance
    c. Automation
3. Principles
    a. Independent, linkable, addressable and complete

The Biomedical Concept Layer

As outlined in the previous chapter, a significant issue with the standards is their siloed nature. Across the current standards, we find that individual atomic data items, the variables, are replicated. This results in a need to map from the equivalent items when moving data across the silos. There is also a danger that one standard changes its definition while the others do not and we are being forced into unnecessary extra work.

A simple example is the notion of age. In a protocol we specify that we require the collection of Age. In the collection phase (CDASH) this gets standardised as AGE and AGEU. We then repeat the same definition in tabulations and SDTM which is then again repeated for ADaM and the analysis step. This is illustrated in Figure 2-1 below.

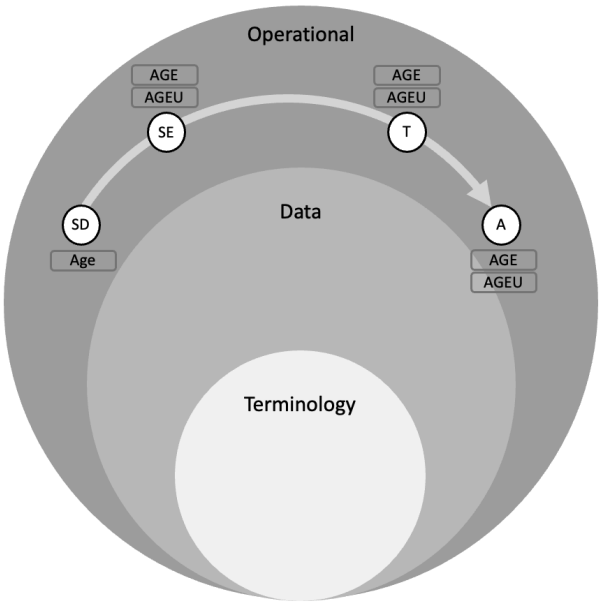


Figure 2-1: Repeated Definitions

Within Figure 2-1 SD = Study Design, SE = Study Execution, T = Tabulation and A = Analysis

Another important issue with the standards depicted in the figure is we can see that we have few layers in our standards. We have the terminology layer and the operational standards (CDASH, SDTM, ADaM) layer but, effectively, everything is compressed into a single layer, the CDISC standard.

If an intermediate data layer is introduced [Ref 3] within which we define the BCs and define the standards based on those BC definitions, we can remove the silo effect. This is depicted in Figure 2-2. We define the unit of knowledge once and then reuse across the operational standards

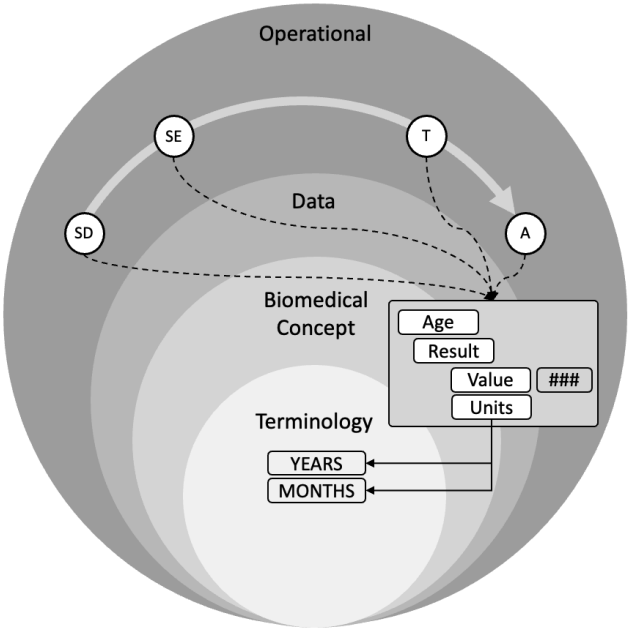


Figure 2-2: Introduce a BC Layer

As can be seen in Figure 2-2, the BCs refer to terminology while the operational standards refer to BCs, the "flow" of references is Standard -> BC -> Terminology and not the other way. Terminology knows nothing about the BCs and BCs know nothing about the standards. This is an important principle. The science does not change (the BC) but our standards may. It would also allow for multiple standards to be maintained for different use cases.

The BC becomes a new standard and is reusable across operational items, be they forms, domains, electronic data loads, wearables etc. The BC layer becomes future proof irrespective of technology as the definitions are based on the science of the observation, not the means of collection or subsequent tabulation or analysis.

A key benefit is that the relationships from the operational items allow a permanent link, or relationship, to be established between standards thus allowing for automation. This link removes the need for "mapping". Automation brings consistency and improved data quality. What this demands is that we have a solid design to which BCs conform, such that the links / relationships can be formed.

The word "Mapping" is used widely within the industry; we seem to spend our lives constantly mapping. A definition for mapping that seems appropriate to the industry is

> *an operation that associates each element of a given set (the domain) with one or more elements of a second set (the range).*

A mapping is a relationship and, we as an industry, need to map because the relationship is missing. This is an important point, mapping implies we have not defined something up front.

We need to define these relationships and not be restricted to simplistic 1 to 1 relationships such as the use of variable names. We need to ensure those relationships are there from the start, remove the need for "mapping" and enable automation.

The BC, the unit of knowledge, becomes the “glue” between the standards. Figure 2-3 illustrates this point with the example of a form and a SDTM domain. Here we would normally annotate the CRF with SDTM annotations, be it a PDF or some electronic form, and then perform the mapping in code because we do not have those relationships within the machine. Now we can use the links from the form to the BC onwards to the domain to allow the machine to do the work for us.

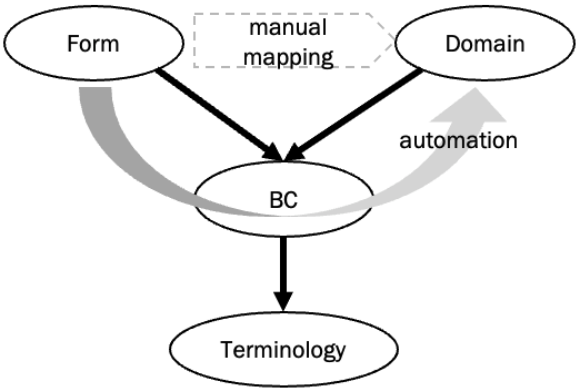


Figure 2-3: BC as the Glue

However, we need to be careful. Care must be taken so as not to disrupt our current world, we need to preserve what we have; BCs and our existing world need to work side-by-side. We should, however, remove the “wrinkles”, those elements of the standards that are not working, don’t fit the pattern etc.

With the introduction of the BC layer we can now begin to see our world structured as a series of sub-models and layers. In Figure 2-4, this has been drawn with the operational elements across the top from Protocol and Study Design on the left through to Analysis on the right. Below is the BC layer with references from the operational layer to the BC layer. The BC layer then refers down to the Terminology Layer.

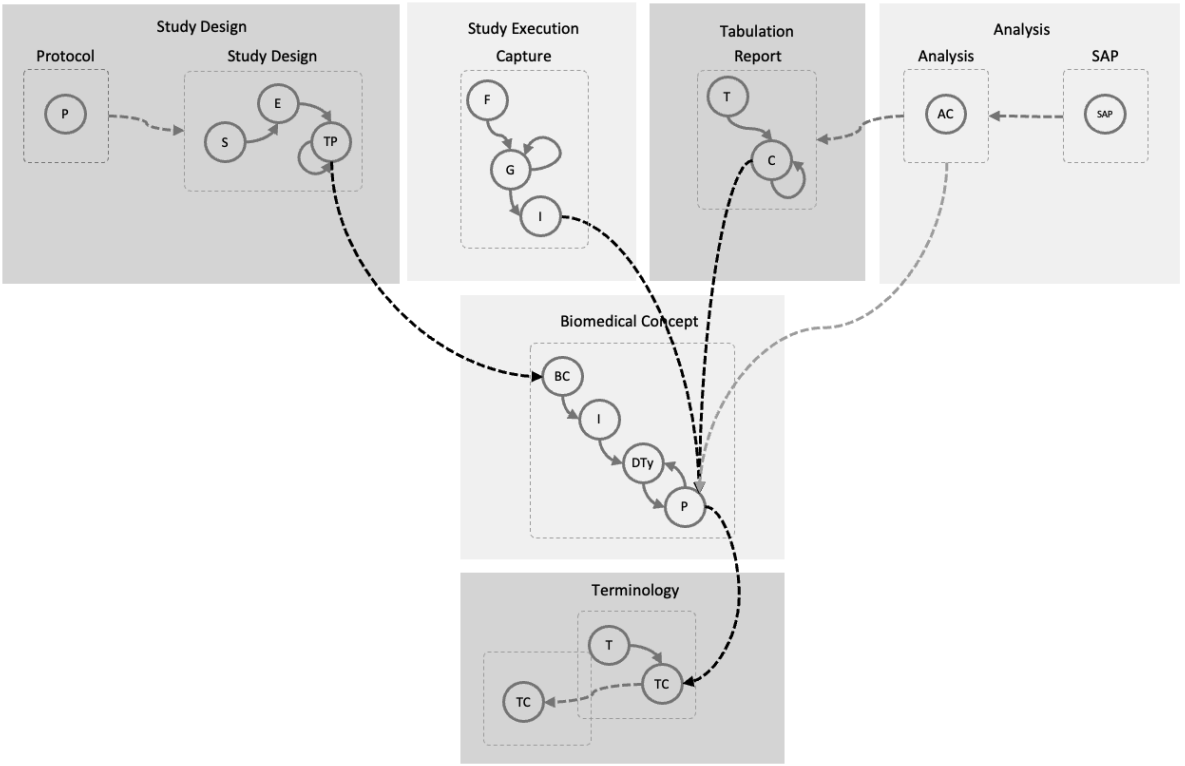


Figure 2-4: The Layers

Summary

1. Relate our standards to a consistent view of the data
2. Remove silos by using the BCs to link across them
3. A consistent design for BCs to allow for the automation
4. Layered approach: Standards -> BCs -> Terminology

Design

Overview

This chapter details the design of the BC, the BC templates needed to standardise BC content and a Canonical Model that drives the construction of the BC templates and enables the automation so desired by the industry. The Canonical Model is a key enabler that allows not only BCs to function but facilitates interoperability with other models.

Initial Definition

We can start by providing an initial BC definition

> *A Biomedical Concept is a computable specification of the data points of a single specific clinical recording excluding the context in which the recording was made. As such it is an atomic definition that is uniquely identified and addressable,*

Context

As stated above a BC does not include the context of the recording. A BC is a recording that needs to be reusable in a number of circumstances. Obviously the main one is a clinical study but other contexts such as healthcare, public health are equally valid.

We also need to consider the relationships between the context needed and a BC. Typically we wish to associate a recording with the person to whom it relates. But of course there will be many recordings per person and that person can be viewed from one or more perspectives as noted above.

From these initial thoughts we can place the boundary between where BCs end and where other ideas or notions start. This is illustrated in Figure 3-1.

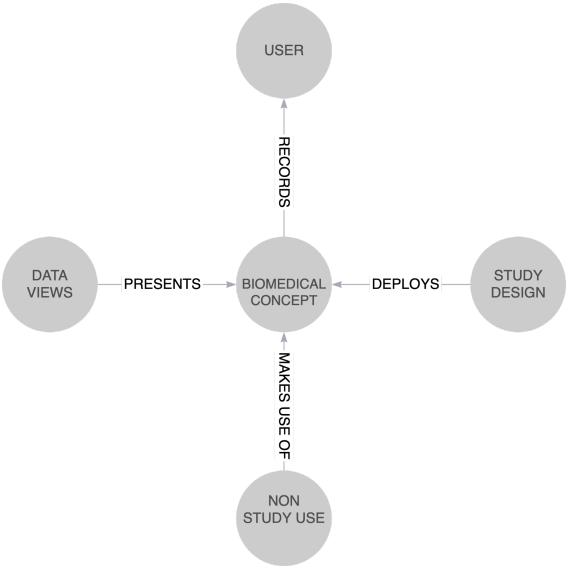


Figure 3-1: Context

What is a Biomedical Concept

BCs are models to structure the data resulting from a recording on a subject. At the most basic level we can consider a recording to consist of

1. Identification - a means of identifying the actual observation.
2. Result - the result. This may include a null flavour, a null flavour being an indication that the desired data are absent and the reason for it not having been collected.
3. Qualifiers - any further data that is needed to understand and qualify the result.
4. Timing - when the result occurred, be it a point or a range.

Additionally we might add

1. Comments - Any comments noted at the time of capture.
2. Categorisation - Impose some classification on the observation.

At its most basic, A BC could be seen to be as depicted in Figure 3-2, an identifier, the result, the timing and the qualifiers bound together as an addressable and indivisible piece of knowledge.

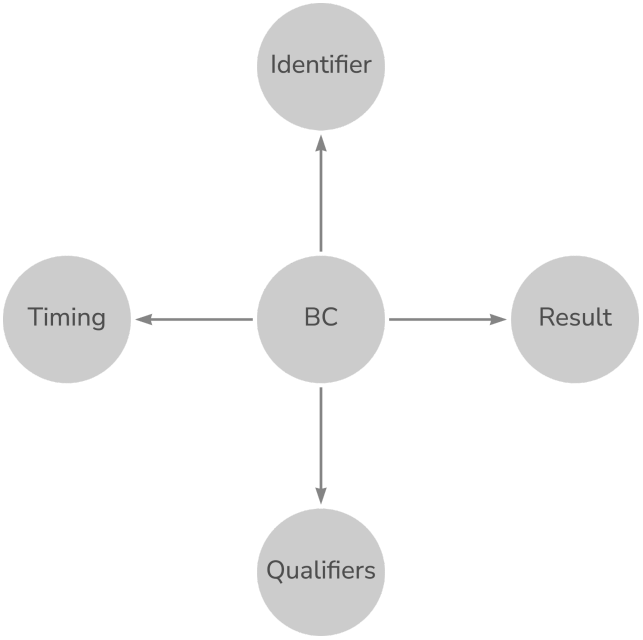


Figure 3-2: Simple Notion of a BC

Why do we want to draw such pictures? The answer is that if we understand the nature of the data we collect, the observations we make, we can better structure [model] our world. A clearer and more precise understanding will allow us to build the necessary relationships into our data from the start. As was noted earlier, if we have relationships from the start of the life cycle we remove the need for mapping later in that same life cycle.

Over the last few years, Armando Oliva has been thinking about the nature of these observations and has documented his thinking in his blog [3]. Inspiration for what is presented below is based on his thoughts, as well as the current SDTM standard, experience, and various thinking and experience of use of BCs while putting the observations into the context of a study. Figure 3.3 is based on this combined experience.

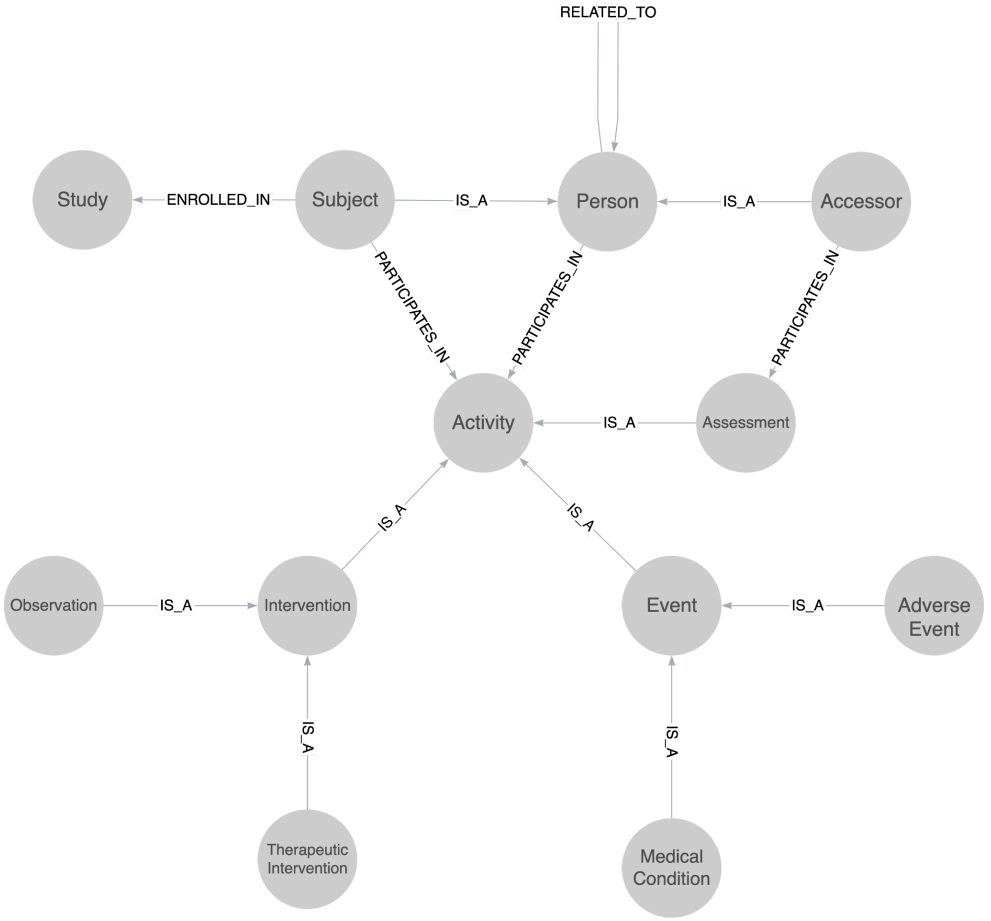


Figure 3-3: Context

Figure 3-3 leads to a different definition of a Biomedical Concept from the initial one presented earlier.

> *A Biomedical Concept is the recording, in data, of a single activity within a clinical study*

Now this definition requires a definition of an activity within a clinical study but does provide for a shorter and crisper definition. Keen observers will also note that the above does not align with current SDTM thinking of Finding, Intervention and Events classes

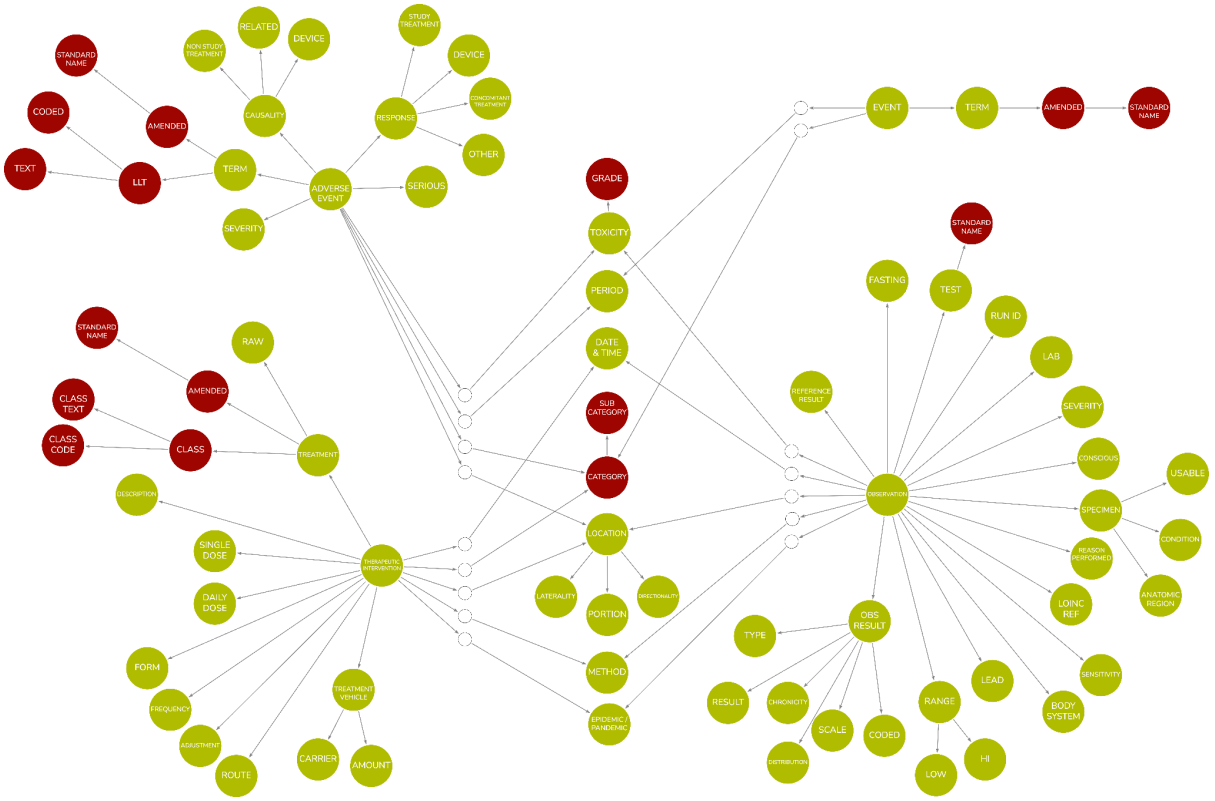


Figure 3-4: Canonical Model

Note: Colour coding, red = derived data items.
Note: Assessment also needs to be added to the model

The above model is preliminary, that should be stressed. Previous implementations have used a simpler Canonical Model but it is believed that such a model will provide a more complete view of the recordings undertaken in clinical research. It needs further refinement and work but is included, as it provides a fundamental piece of the BC solution.

The above model details the set of data that make up recordings. Each leaf is a complex data type. Many are coded values but others may be physical quantities, addresses or simple strings. A complex data type model is needed and HL7 provides such a model. The HL7 FHIR data types, see [5], offer a suitable model and would provide commonality with healthcare and not reinvent the wheel.

The combination of the model and the complex data types provides for a richness in relationships. Each leaf is intended to have a unique identifier that can be used to reference the unique concept: the value of a recording, the units related to that recording, a qualifier that specifies the method used to make that recording. The unique reference can then be used by any structure holding the data of the recordings to state “I am an X”. An X in another format can then be aligned or equated.. The hope is that the canonical model provides a means by which different data formats can be related and conversion automated.

Previous experience and implementations attempted to use the BRIDG model as the Canonical Model. This did not prove to be easy, as the BRIDG model is complex and use was sparse as we are only concerned with the data structure, not all the control structures found within the model.

This previous work also used ISO21090 data types but the complex nature of the datatypes caused issues, as did the recursive nature of the definitions. The healthcare data types do work but a pragmatic approach does need to be taken, hence using the FHIR data types.

These models provide a canonical representation of our data and an understanding of the complexity of what we are recording and the relationships within those recordings. But why do we want this precision? This precision provides for:

1. A method by which we can define templates for our recordings so that we can bring consistency to our recordings across the industry and thus drive data quality and utility
2. A better understanding of our data in that we define up front, the structure of our data and the relationships inherent in our data.
3. Ability to extend our recordings to accommodate new science - new data - or to overcome operational issues - extra data - but doing so from a position of knowledge on the structure of our recordings such that we can do so in the best possible manner rather than a casual creation of another supplemental qualifier.
4. The canonical model will also provide a mechanism to link to other data models thus allowing for integration of other data sources into clinical research such as Real World Dtaa. This is exploited further in a later chapter.

Biomedical Concept Model

General

Given we now have the above canonical representations, we can now generate a model for a BC, a model that incorporates the ability to handle the recordings we wish to structure.

Many BCs will have a similar structure but there will be variations, for example, consider a basic vital signs test versus laboratory tests with the extra information that is captured. This gives a need to have a set of templates that:

1. Provide a consistent subset of the Canonical Model
2. Provide a more machine friendly structure and implementation
3. Provide the links from an instance of a BC to the Canonical Model
4. Make use of complex data types as not all our data is simple numeric values or a coded value set, some are complex such as geospatial data

The design used for the template and the actual concepts is the same, a concept reflects its template with the template providing the link to the Canonical Model.

The design is shown in Figure 3-5 below

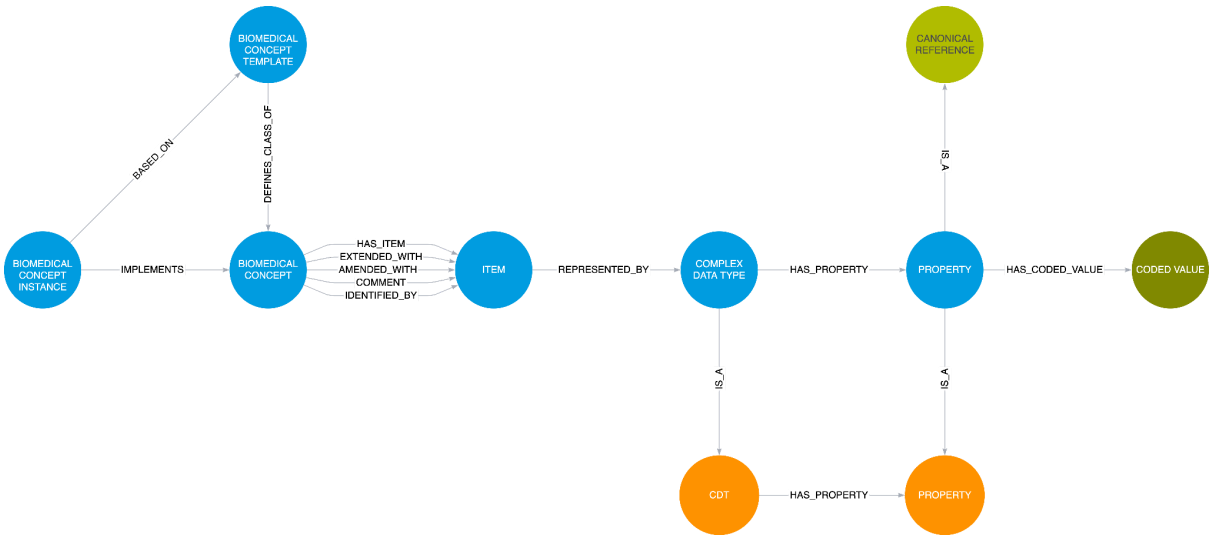


Figure 3-5: The BC Model

The model contains the template and the instance which both employ the core BC model. At the head of the model is a root node that binds the entire definition. A BC is composed of 1 or more Items with some of those items performing specific roles:

1. Identifier - The item that will identify the BC as a whole.
2. General Collection - The set of all items for the BC
3. Comment - A comment item. The comment is global to the BC as a whole.
4. Extended - An item that is an extension to the BC. This is for planned additional data and should be reflected in the template prior to deployment.
5. Amended - An item that has been added to the BC during data collection. This is to allow for Ad Hoc extra data; a “get out of jail for free” card!

The model is based on experience of implementing several versions, from an early Excel-based version to two graph-based versions. It is expected to evolve, as people become familiar and ideas improve.

BC Template

The BC Template holds a pattern for a particular type of recording, a basic observation, a general lab test, a specific type of lab test etc. It is expected that there will be 10s of templates but templates can be defined at any time. Such a definition should be version managed thus allowing changes.

The template is there to define the set of items within a BC and the data types associated with those items. A template does not define any content but defines the structure of a set of instances.

An item may be able to have more than one result data type, it may be coded, a value and units, free text. Templates must be able to accommodate this and allow for the selection when a BC instance is constructed. Using vital signs again as an example, consider a simple

template that has to accommodate Height with a quantity result versus Frame Size with a coded result.

Results may also be complex structures in their own right, such as addresses, geo data etc. This is one major advantage of a BC representation, we are not constrained by a rectangular form and can thus have greater freedom to represent the data we record accurately.

The following figure depicts a BC template, albeit simplified to reduce the number of items in use to make the figure readable. Each item is linked to a leaf in the Canonical representation via a complex data type.

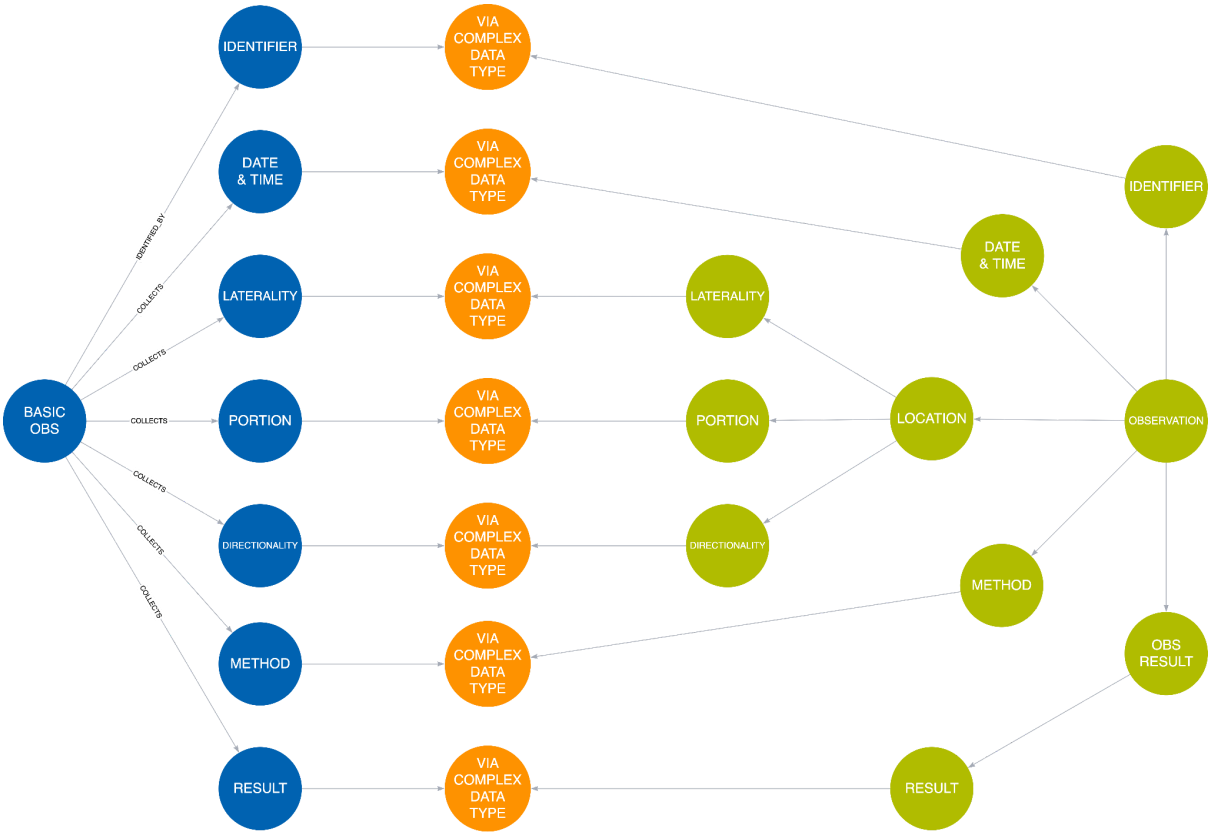


Figure 3-6 Example Template

One item is expanded to show the action of the Complex Data Type and how it is used by both the template and the canonical representation with each leaf being connected to the appropriate matching leaf.

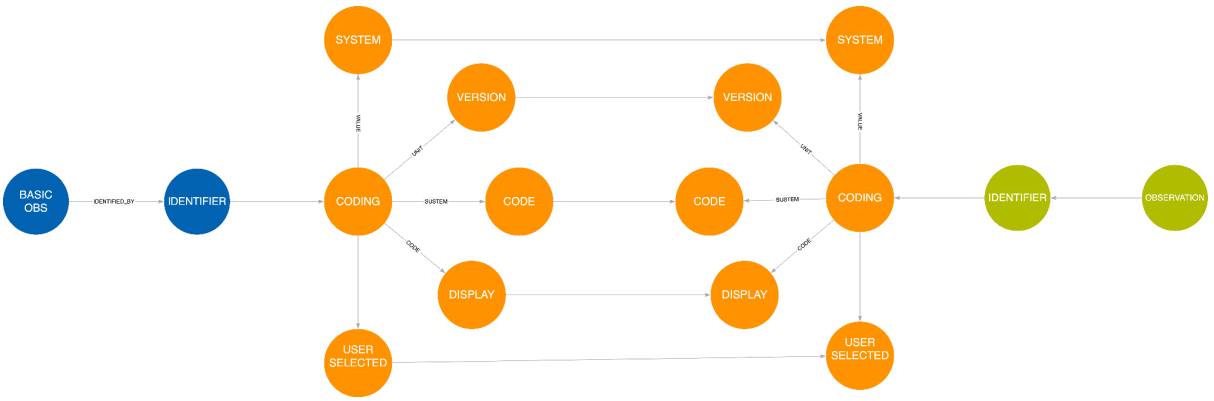


Figure 3-7 Expanded Item

One important concept to note is that each leaf on the canonical side can be referenced by multiple templates, there is only one observation.identifier.coding.code reference in the Canonical Model, there may be many references to it.

Reiterating a point from earlier. This may look complex and in some ways it is but we should not shy away from complexity but embrace it. It should be remembered that each leaf in the canonical model is intended to have a unique reference. Thus the leaf of the BC template is simply quoting that reference, stating “I am an X”. That unique reference might be a URI, a GUID, the simple string we used in the previous paragraph “observation.identifier.coding.code”, as long as it is unique that is all we need for an implementation.

We now have a Canonical Model detailing an observation and BC Templates that have precise relationships with that model

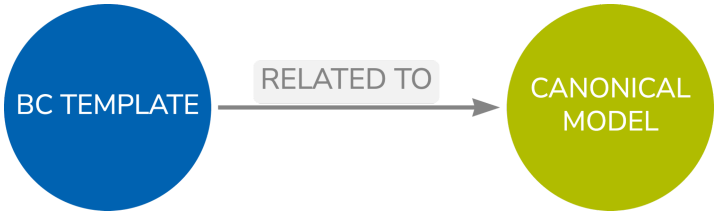


Figure 3-8 Model Overview One

BC Instance

A BC Instance holds the precise definition of a single recording. The purpose of a single BC is to provide a detailed definition of a recording including terminology references. Such a definition should be version managed thus allowing changes.

A BC instance is based upon a template which is constrained to what is required. A template may define a method by which the recording was captured but the BC may not need the method to be recorded. Certain items, such as the identifier and the result, will always need to be defined and this drives the need for a template to define mandatory and optional items.

Below is a simple example modelling the CDISC Height BC. The BC has been simplified to illustrate the essential concepts to keep the figure readable. The BC is based upon a template and the structure of the BC instance follows the template. The BC Instance is linked to the template at two levels, at the top most level where the instance is linked to the template and at the leaf level such that each data type property is linked from the instance to the template; one example is shown in the figure below for Date & Time. The BC instance also defines the terminology thus forming a complete definition of the observation.

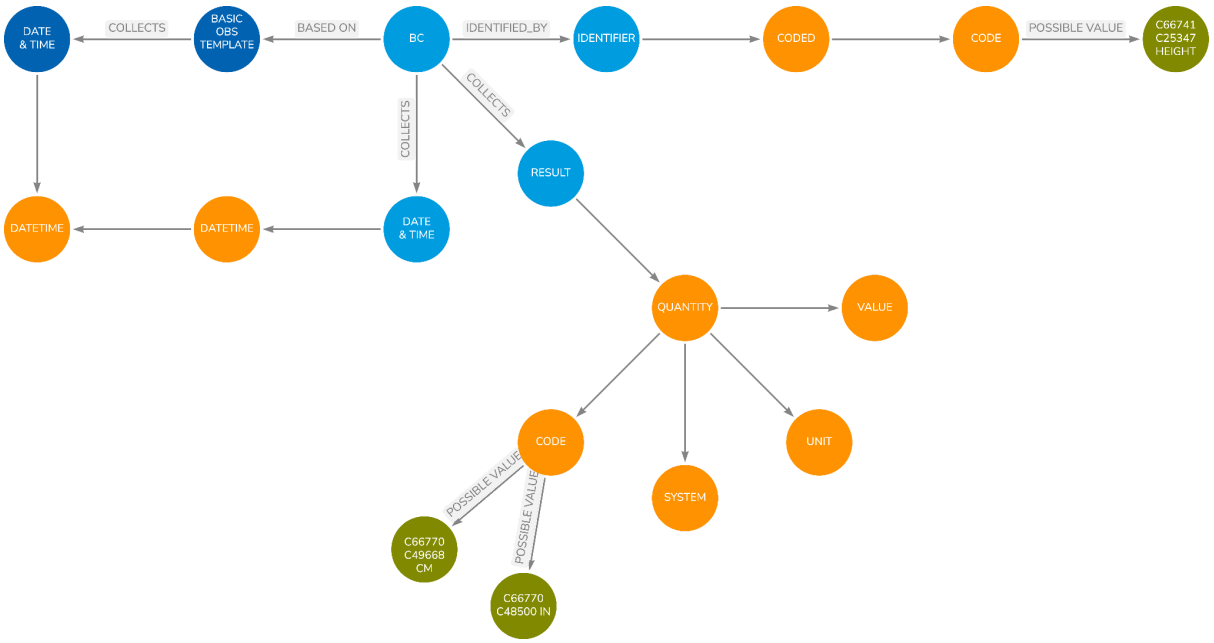


Figure 3-9 BC Instance Example

We now have a Canonical Model detailing an observation, the BC Templates that have precise relationships with that model and then BC instances formed from the templates. Note that, by linking the templates to the Canonical Model, the instances inherit the linking so that all the relationships in the canonical model are available to system processing BC instances.

Figure 3-10 Model Overview Two

Collections

There will be a need to bring BCs together into collections. Collections may take several forms:

1. A collection of BCs that forms a higher level BC or logical entity such as Blood Pressure formed from Systolic and Diastolic components.
2. Pre-determined sets of questions such as questionnaires or sub parts of a questionnaire.
3. Laboratory tests group into a panel or other such collection.
4. Forms but forms use several BCs to allow for the collection of data. CRFs are a very specific use case.
5. Tabular structures such as data transfer specifications where data content is passed from system to system

Summary

1. Context
    a. The boundary of the BC in relation to the rest of the world
2. Canonical Model
    a. A generic model, applicable to clinical research, detailing the nature of the observations we make and the relationships inherent within our data
    b. Canonical model employs data types to manage the lower-level relationships
3. BC Templates
    a. Definition of the data items that form the key content of the observations we typically make
    b. Based on the Canonical Model thus inheriting the relationships and thus the knowledge
4. BC Instances
    a. Precise definition of an individual recording.
    b. Complete terminology specification.
    c. Place data at the centre.
5. Independence, Structure and Precision
    a. BCs are totally unrelated to existing CDISC structures except for terminology.
    b. Better structure existing CDISC content and fill the gaps.
    c. BCs are precise.

Linking to SDTM

Overview

We now have a BC model connected to a canonical reference model. So what is next?

SDTM is of primary importance to the industry as it is a major part of a regulatory submission. One major aim of BCs is to aid automation. So how do BCs help with SDTM and automation? This chapter will explain how we can link the BC and SDTM world to allow for that automation.

SDTM and BCs

SDTM is a combination of three types of data: a) the raw data as collected, b) data derived from that collected raw data and c) a set of timing information derived from the raw data and the study design.

The BCs are designed to better structure the raw captured data and thus the linkage between BCs and SDTM only concerns the raw data fields. The mechanism for linking the SDTM to the BCs is via the Canonical Model. This decouples the BC world from the SDTM world and allows for development to proceed on either without a ripple effect when one or the other changes.

SDTM is simply a rectangular structure in which each column is an atomic data item. In essence, SDTM is the placing of the Canonical Model into a rectangular form with the addition of the derived and timing data.

Consider the original result and units, these obviously map to the value and units in a quantity value or just to the result of a coded response as there are no units involved. The method maps to the coded response in the method item. We can simply link the variables in SDTM to the respective leaf nodes in the Canonical model.

So as to make the relationships as generic as possible, it is sensible to link the SDTM classes and the variables therein to the Canonical Model. By doing this we increase the flexibility as the relationships can be inherited by the domains and variables. This is the same notion as the BC templates linking to the Canonical Model.

The following figure shows a few examples to illustrate the relationships. Here --ORRES is linked to the various leafs in the Canonical Model that relate to an observation result. Obviously a result can be coded, could be a quantity and thus there are several routes available. The --METHOD variable is also linked but that is only ever a code value.

Note that the class variable definition is linked to the leaf data type node in the canonical model. Remember that the canonical leaf node has a unique reference, the same that is being used by the BC templates. Given that the BCs are also connected to the leafs of the

Canonical Model we now have full linkage between BCs and the SDTM model simply through the use of the unique reference. Our world of knowledge and connectivity grows.

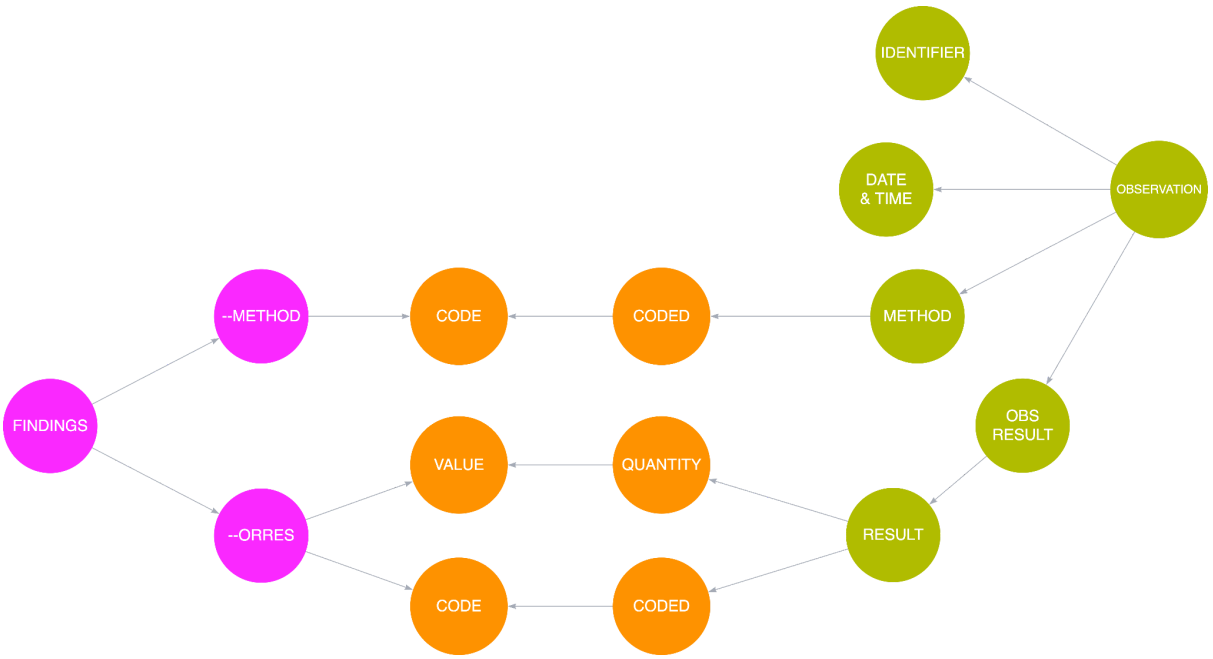


Figure 4-1 The SDTM Canonical Model Relationship

It should be noted that certain BCs and the templates will link to certain parts of the Canonical Model as will the SDTM models and this reflects the variation in our data. The templates will dictate what BCs can go to what classes within SDTM but it should be noted that the BCs are linked to classes and not individual domains. That allows for a BC to be placed into many domains thus providing greater flexibility in the future. It also means we can allow a BC to be placed into multiple domains and the exact choice left to the study as to how best structure the data for the science.

We can also link all versions of the SDTM model to the Canonical Model and thus link the SDTM versions. This will be discussed further in later chapters but this does facilitate automated conversion of data from older versions to a newer version of SDTM.

The linking of the Canonical Model to SDTM means that we have also, effectively, added more relationships into the SDTM model, those of the canonical model, that a machine can understand without disrupting the current SDTM model.

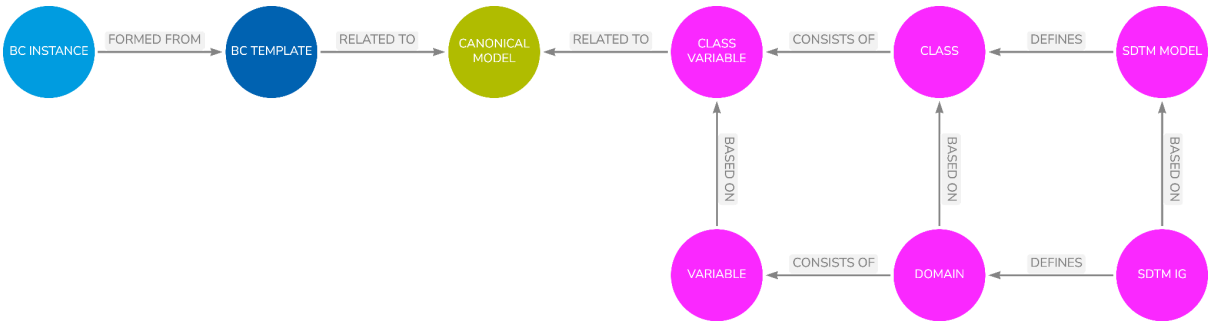


Figure 4-2 Model Overview Three

Issues

As a result of linking the SDTM and a Canonical Model starts to expose a few questions and issues with SDTM. These include:

1. The use of --CAT and --SCAT variables and the desired use within domains. The use of these variables has been shown to be inconsistent and it maybe worth looking into best practice for them
2. The set of variables used to record when observations have not been captured during a study the reason for the failure to collect can be better modelled within the BCs, for example, with the use of null flavours. This better model can then be related to the existing variables.
3. Important flags such as fasting, clinical significance, epidemic / pandemic is another area where some better practice may be possible.
4. Comments, both at a BC level and an item level could also be improved using BCs and better mappings into SDTM be found.
5. One significant advantage of the approach is the addition of supplemental qualifiers. The Canonical Model offers the opportunity to provide better guidance to the community on the addition of Supp Quals and their addition to BCs and SDTM. With better visibility of a model SDTM practitioners will be less inclined to just add a Supp Qual in a haphazard manner but rather make more considered assessments of where a Supp Qual should be placed.

Summary

1. The SDTM and the collected raw data is a rectangular form of the Canonical Model
2. By linking the SDTM to the Canonical Model we
    a. Add relationships to SDTM
    b. Link to BC Templates and thus BCs
    c. Decouple BCs and SDTM
    d. Provide flexibility into which BCs can be placed
    e. Allows for BCs and SDTM to have independent development paths
3. Once the SDTM model is linked to the Canonical model all BCs will be compatible with the SDTM,the model ensures compliance

4. Canonical model provides for much better understanding of SDTM and better decision making in the placement and use of Supplemental Qualifiers

Protocols, BCs and Digital Data Flow

Overview

This chapter details how BCs can be used as part of a protocol and linked to the corresponding study design such that the precise study data needs - the study data contract - can be formed. This study data contract can provide a precise and implementatable specification for a clinical study.

Study Design

A study design found within a study protocol document typically defines a design at a number of levels:

1. The arms and epochs which define the study cells (intersection of an arm and an epoch) which link to the elements (reusable cell content) and the intended treatments.
2. The Schedule of Assessments where high level design links to the intended visits and the procedures and assessments needed to prove, or otherwise, the scientific hypothesis being put forward
3. A further level of detail on from the SoA on the data to be collected

Currently, this is very much a paper exercise with protocols being delivered in a PDF form. Tools and designs are emerging that provide machine readable designs with Transcelerate's Digital Data Flow (DDF) being at the forefront of such work. These initiatives address the higher two layers but have yet to provide industry with the models needed to provide a full, machine readable design that provides a complete study data definition and the data contract.

BCs can assist with the lower level and when attached to the higher levels, can provide a complete study design definition.

Protocol and Observations

Within many protocols we see examples of BCs without realising it. For example, we see text such as the following:

> *Demographics … will be recorded*
>
> *Visit 1:*
> - *Age*
> - *Sex*
> - *Race*

Here Age, Sex and Race are BCs, units of knowledge that are composed of several parts that we dont really want to sub-divide. Another example is:

Vital signs (Systolic Blood Pressure and Diastolic Blood Pressure, Heart Rate) ...

Again, Systolic and Diastolic, Heart Rate are again, slightly more complex BCs but again BCs. In many Schedule of Activities (Assessments) we see footnotes listing observations to be undertaken at baseline visits and a subset of those BCs for subsequent visits. Again, these talk in terms of what are really BCs.

We also see collections of BCs referenced, such as:

- *"Blood chemistry includes measurement of …"*
- *"... standard haematology tests [including haemoglobin], blood chemistry tests [including LFTs] …".*

The BCs are these observations referenced from within the protocol, either individually or as a collection. The references in the protocol are simply the meaningful, human readable, names of the BCs instances.

To build the detailed study design we need to provide a mechanism to link individual BCs or collections of BCs to the higher level study design structures. If this is achieved then we can provide a complete study definition down to a detailed data level; we can build the study data contract, the data needed to prove, or otherwise, the hypothesis.

Data At The Centre

The set of observations within the design, be they detailed individually or as a collection (e.g. a laboratory panel, a questionnaire etc), forms a precise Data Contract that needs to be met by the data collection process. The Data Contract is focused on the data, it is a data template, and does not care, as yet, about the means of capture.

By building the data contract we place data at the centre [6]. We need to define the data contract irrespective of the data source. Some sources are important, such as validated instruments and patient reported outcomes, but laboratory data might arrive via a CRF or a tabular electronic data load, but the nature of collection is less important.

We are also splitting the data from the presentation, something we in clinical studies are very good at merging. The data are the data, how we collect those data is important for some of the content but much less so for the vast majority. Here the study design defines the data and we can move away from thinking about CRFs and the "how do I capture the data" to what the protocol should focus on and "what data is needed".

Endpoints and Objectives

In a transcelerate document issued as part of the DDF Hackathon in November of 2019 appeared a paragraph. It linked the notion of Endpoints and BCs

For example, using the TA Library for Asthma, a study in severe asthma could have as its Primary Objective "To evaluate the effect of drug x in participants with severe asthma." The primary endpoints linked to this objective are limited to "absolute change in percent of predicted FEV1 from baseline to [Week X]" OR "increase [magnitude of change] in FEV1 from baseline to [Week X]." This also implies that the FEV1 biomedical concept will require spirometry assessments to be scheduled at baseline (CDM: primary timepoint) and week X visits (CDM: secondary timepoint), and that FEV1 measurements will need to be captured in the study database, either by EDC or via data transfer. Further, options for Secondary Objectives include FVC or FEV1/FVC ratio (spirometry), reduction in symptoms (questionnaire data) or fewer Clinical Exacerbations (medical history or diary data) or reduction in the use of rescue medication (diary, dosing device or medication count data). As each objective is chosen, the appropriate choice of linked assessments and measures would also be assembled in the tool using the latest available standards for that assessment.

This highlights the linking of Endpoints, in this case change from baseline, to two instances of a BC, in this case FEV1. But what is interesting as well is not only the link from the protocol objectives and endpoints to BCs but also the connection to the timing aspects of the study.

This leads to a rethinking of the structure of a machine readable design with the focus being placed on the timing aspects of the study.

Timeline Approach

The timeline approach is designed to represent the study more as a timeline reflecting the study high-level design while placing the timing information in a study into a single location within a model. All other entities that reflect the timing then refer to those centralised timing structures. This results in having a single source of study timing rather than spreading the information around the model in a number of places.

Consider figure 5-1 below. The core of the model is a Time Point Node. This node represents a point in the study at which something needs to happen, either some data collection or a procedure(s) needs to be performed. Each Time Point is related to an Epoch, an Arm and a Visit such that its relationship with the high level design is clear. However, the important timing information is maintained within the Time Point nodes thus keeping the timing in one place. The respective timings for a visit etc can be determined by inspecting (querying) the Time Point nodes.

Walk through Figure 5-1 in the next draft.

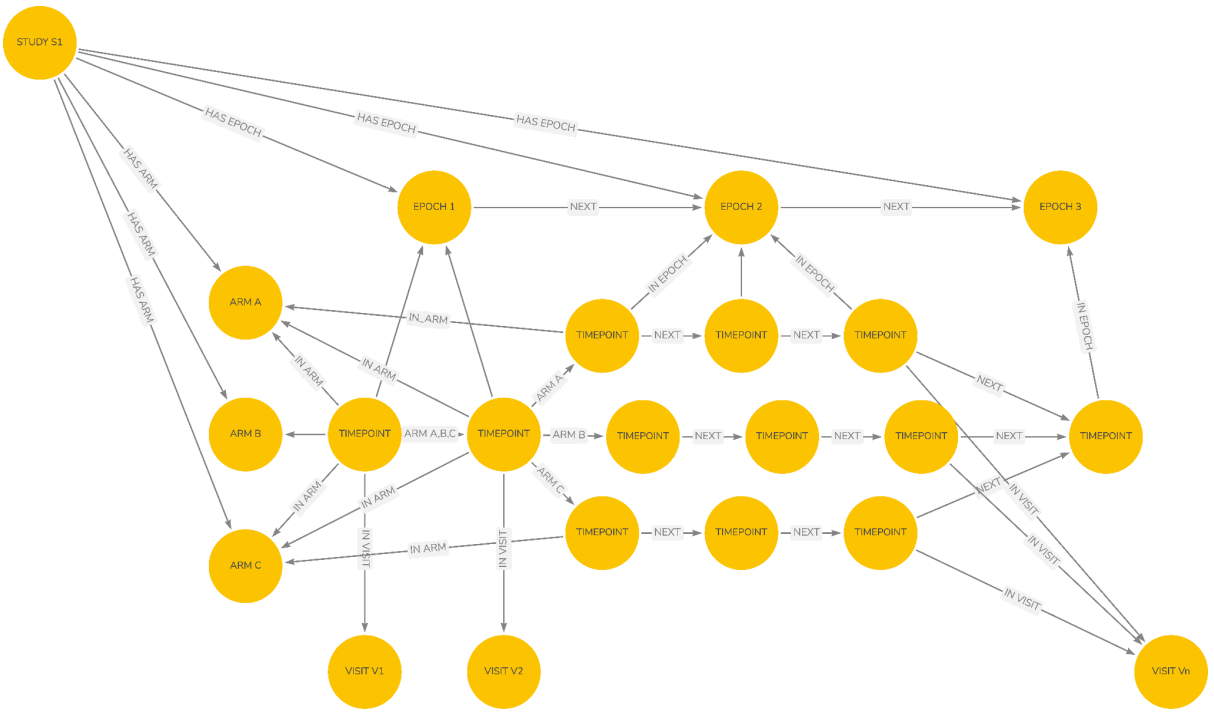


Figure 5-1 Study Design Using Timepoints

Figure 5-2 takes a closer look at a Time Point node and their relationship with each other. Each Time Point node defines the desired actions for that point in the study and links to the next Time Point. The links between Time Point nodes will require some timing and logic so as to determine what, if any, rules apply to the transition from one to the next. This allows for multiple paths and would also allow cycling back to earlier parts of the study if so desired.

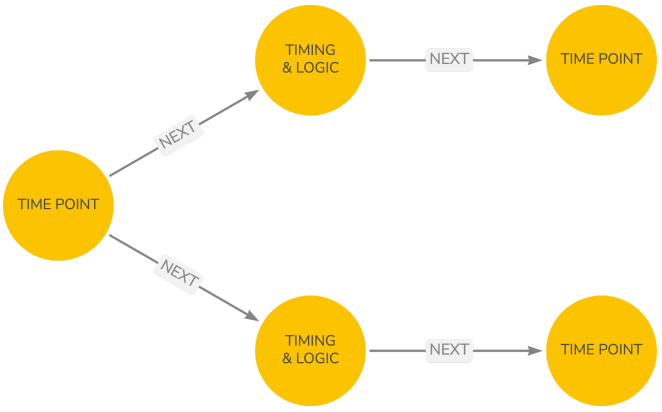


Figure 5-2 Time Point Logic

Figure 5-3 illustrates the specification of the actions required at each timepoint. This will consist of a sequence of BCs, collections of BCs (e.g. a laboratory panel) and Procedures required. If necessary timing can be provided between the BCs, for example perform a procedure and then measure Heart Rate every 5 mins for 30 minutes, using the same type of logic as between Time Points.

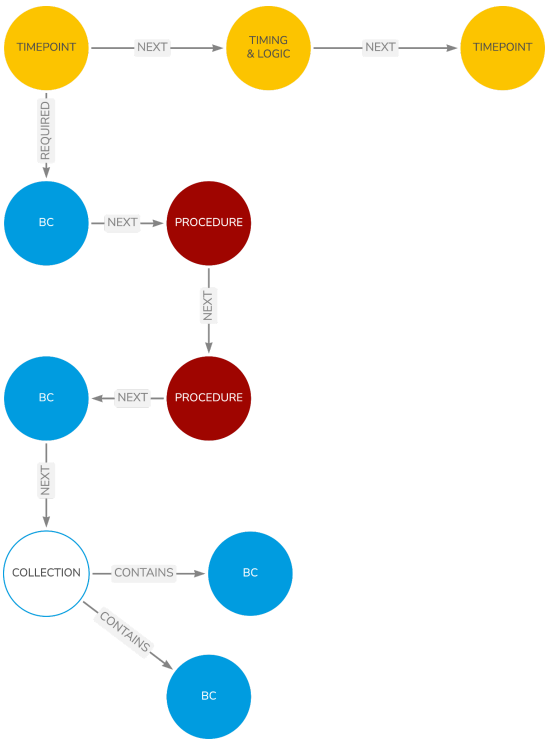


Figure 5-2 Time Points

We begin to see a structure emerge, a series of timepoints, able to reflect the epochs and arms within a study, linked to the arms, epochs and visits with each time point detailing the data contract for a study, be that BCs or collections thereof. This now provides the study design precision needed for automation in downstream systems. Figure 5-4 below illustrates several Time Point nodes each with the required data collection.

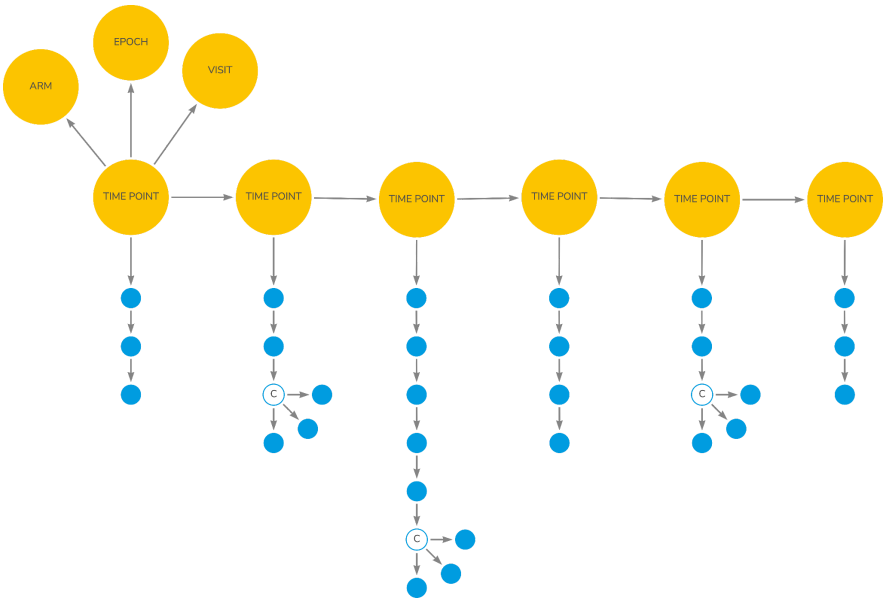


Figure 5-4 The Data Contract

Returning to the issue of endpoints, Figure 5-5 shows an endpoint linking to two BCs representing the change from baseline example noted earlier. This echoes the earlier comment about being able to add endpoints and associated data needs to the timeline but also suggests that other patterns such as safety data collection could also be candidates for adding to a timeline. Safety data, baseline visit requirements etc are just data patterns that can exist as a set of BCs with associated timing in the form of a data template. One or more patterns can be added to the study timeline: the objectives and endpoints, the safety needs, the subject data such as demographics.

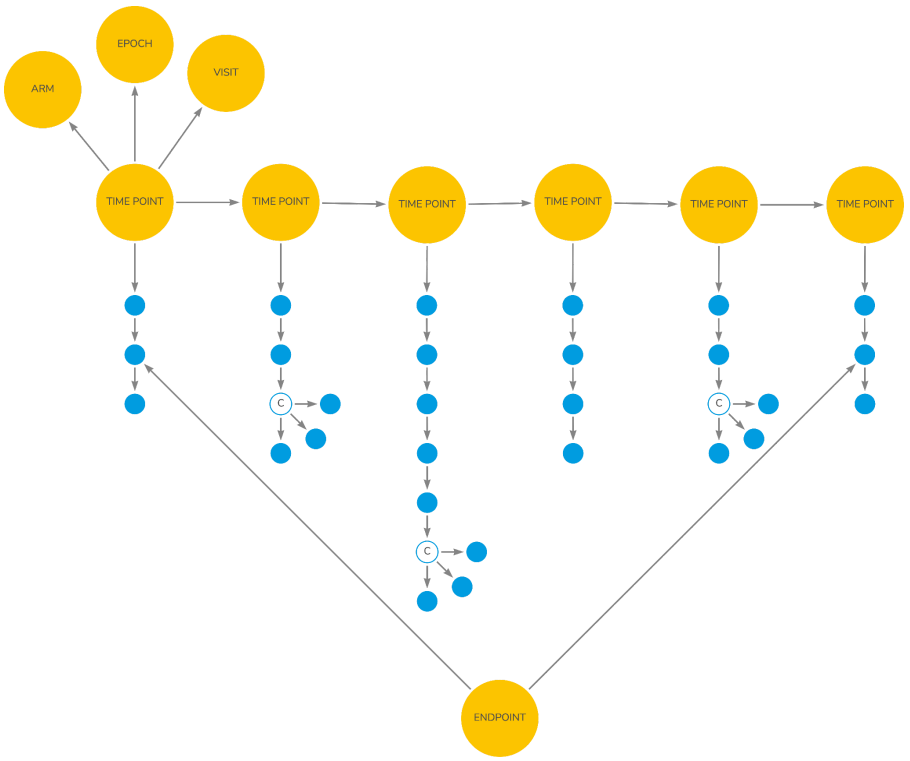


Figure 5-5 Endpoints

Summary

1. We need a study data design that results in a precise study data contract
2. Such a precise design allows for subsequent automation downstream
3. The data contract is a combination of BCs and the precise timing of when the data are to be collected
4. The study timeline is linked to the study Arms, Epochs and Visits.

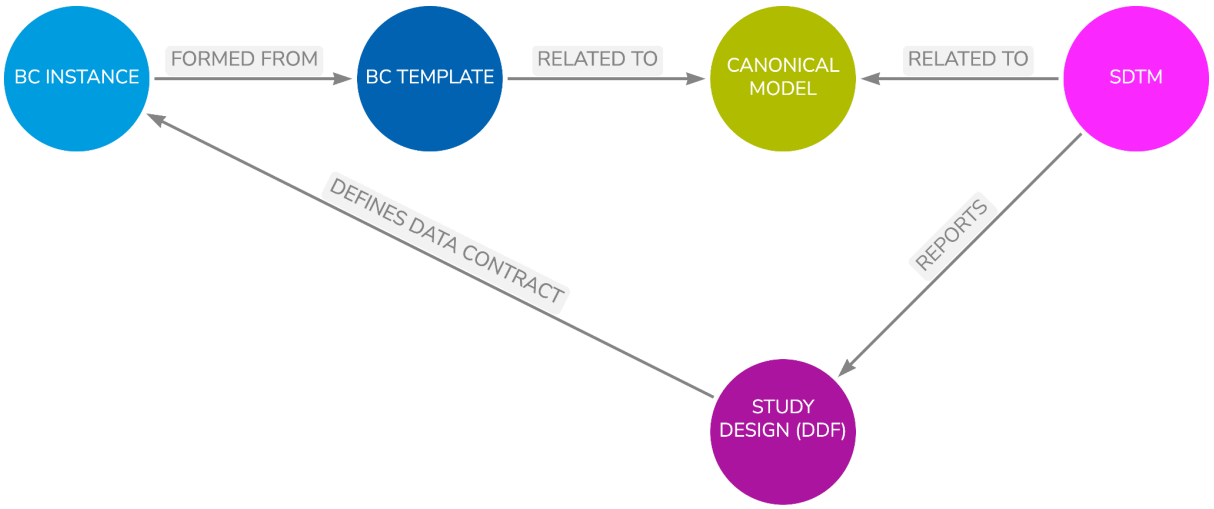


Figure 5-6 Overview Four

Forms And Data Collection

Overview

This chapter will detail how BCs can be used to create standard form definitions.

Much has been talked about standard CRFs since the first ACRO forms were put into an electronic form and the CDASH project was started by CDISC in 2006. BCs work at a data level and, as a result, if BCs are used consistently, the data collected becomes consistent. Consequently, the form becomes less important, the BCs ensure we standardise our data, not how we collect our data.

This then ripples to other forms, we want other data collection to focus on a stream of BCs not on how the data are collected. We want data collection to be a stream of BCs, irrespective of the means of collection.

Study aCRF and Define.xml

Overview

This chapter details how a study annotated Case Report Form and Define.xml can be generated from a study definition that is based on BCs.

Practical experience has shown this can be achieved without much effort from a well organised set of [meta]data that links the Study Design and the associated Data Contract, and the means by which the Data Contract was fulfilled. This combined data can then be automatically presented as an aCRF and define.xml

SDTM Generation

Overview

This chapter will detail the mechanisms needed to provide for the automated generation of SDTM domains using captured data combined with metadata from a study design and BC definition.

See earlier prototyping work covered in reference [4]. This chapter will expand on that earlier work.

Other Data Sources

Overview

This chapter will discuss how other clinical data models can use the Canonical Model to link to BCs. It would be desirable to find automated mechanisms whereby data from such sources as those listed below could be imported to facilitate the capture of Real World Evidence (RWE) data

1. OpenEHR,
2. FHIR
3. LOINC
4. Others TBD

Also, a link to OMOP would also be highly desirable, and the chapter will discuss how this could be accommodated via the Canonical Model.

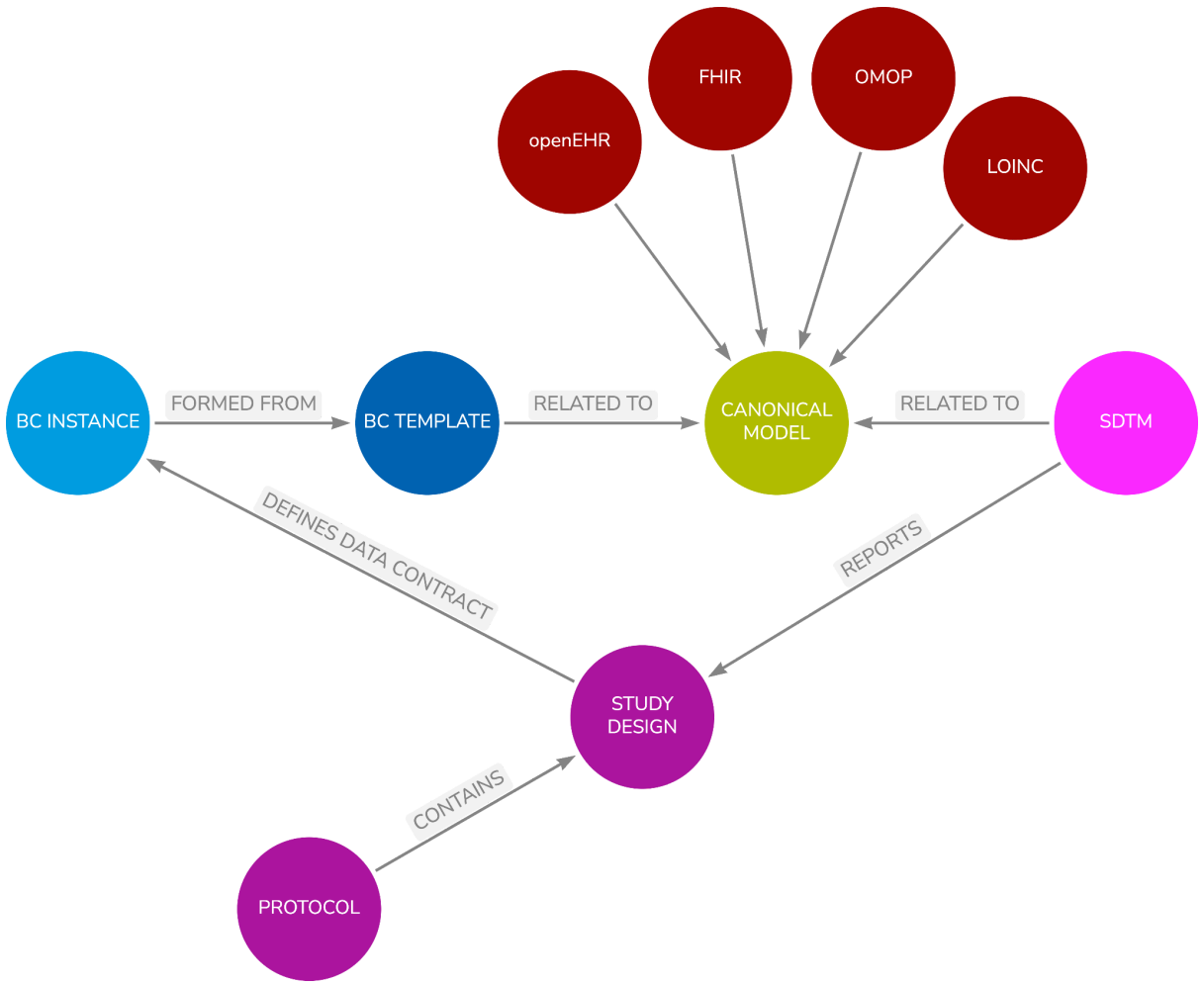


Figure 9-1 Overview Five

Tabular Structures

Overview

This chapter will discuss the use of the Canonical Model with other tabular structures and not just SDTM. Other tabular models can be supported thus allowing flexibility going forward such as SDTM flavours or tailored datasets such as the FDA BIMO site inspection datasets

Example

Overview

This chapter will provide an example of the various concepts detailed within this paper in a demonstration using a graph approach and some simple tools such that people can actually touch the ideas and some data.

BC Mining

Overview

This chapter will discuss how the vast bulk of BCs can be created from existing sources without involving a massive creation exercise.

Formal Definition

Overview

This chapter will detail the formal definition of BCs. openEHR uses a formal definition language, Archetype Definition Language (ADL), for expressing archetypes. This chapter will discuss the need for such with BCs using an RDF and SHACL approach.

The Future and Next Steps

Overview

This chapter will cover any other topics that arise from writing this document.

References

[1] STUDY DATA TECHNICAL CONFORMANCE GUIDE, FDA, February 2014

Note that the current version of the guide is v4.8.1 dated October 2021.
https://www.fda.gov/media/153632/download

[2] RG06 – It's Time to Change, PhUSE US Connect, 2018
https://www.lexjansen.com/phuse-us/2018/rg/RG06.pdf

[3] Blog: Thoughts on Medical Informatics, Armando Oliva M.D.
https://aolivamd.blogspot.com/

[4] Into the Fire, Linking CDISC & FHIR, PhUSE EU Connect, 2018
https://www.lexjansen.com/phuse/2018/si/SI12.pdf

[5] FHIR data types
http://hl7.org/fhir/datatypes.html#2.24.0

[6] SI13 - Removing Silos: Placing Data at the Centre, PhUSE US Connect, 2019
https://www.lexjansen.com/phuse-us/2019/si/SI13.pdf

[7] Digital Data Flow Solution Framework and Conceptual Design Version 1.0, Dated 1 November 2019, Appendix A