Design of a VA Phenotype Library

Andy Zimolzak, MD, MMSc

February 8, 2017

Motivation: VA data elements are not harmonized across stations. *Phenotype* is the term we use for a set of harmonized data elements, or a higher-order concept made of many such "building blocks."[1]

[1] For example, 130 stations' ICD codes + labs → harmonized ICD + labs → diabetes + neuropathy → diabetic neuropathy → which is a subset of diabetes with any microvascular complication → subset of diabetes with any complication at all, etc.

What I imagine for a phenotype library

When a subject matter expert (SME), analyst, statistician, and/or data pull engineer create a phenotype, the result can be: SQL code that represents the SME's rules, or a public database table that represents the enumerated list of included/excluded data items, or even a statistical model of the likelihood of a patient having the given phenotype. Ideally, a robust phenotype library should exist, serving several functions.

(1) The library would make the products of each SME/data review process *available for viewing* by study coordinators or others, with "provenance" (responsible SME and data extractor) clearly noted, to allow study staff to trust a given phenotype. It should also make it easy to browse an overview of all phenotypes without having to drill down into numerous different subfolders. (2) The library would make phenotypes *available for reuse*, for the purposes of other projects. (3) The library would allow easy *searches across phenotypes,* to allow new studies to discover what already exists. Importantly, search should cover the contents of each document—not just titles. (4) The library would make phenotypes *available for renewal,* with their date of creation clearly noted, because data warehouse elements may change over time. (5) The library would *document* the rationale for decisions made by SME, analyst, etc. and allow collaborative editing of this documentation. Ideally, changes would be audited/tracked. Possibly it should allow links among documents. (6) The library should be available across VA nationally.

Operationalizing the library

I think a wiki serves *some* of these requirements well, especially search; and viewing provenance, documentation, and rationale.[2] But I'm not sure whether the current wiki is the best long-term solution. Note that SharePoint has a search feature, but it currently does not work at all.

A wiki does *not* serve some of the functions as well.[3] In my imag-

[2] A wiki is a Web application that allows people to add, modify, or delete content in collaboration with others. Note that functions 1, 3, and 5 involve human-readable English text.

[3] Specifically reusing, renewing, or modifying phenotypes. These functions depend more on machine-readable items.

ined ideal world, people would use Git or other source code management software to store SQL, SAS, etc. code. The main drawback is the learning curve, so I don't know whether this is actually going to happen. Storing code on the wiki is not the worst solution, even though it's nonstandard. I strongly believe that SQL code cut/pasted into an MS Word document will not be long-term viable, and the same goes for "e-mail your code to one person who will be in charge of uploading/managing it."

More about our current wiki: It uses MediaWiki, which is the software that makes Wikipedia run. MediaWiki uses its own markup language and also provides a template system which helps avoid a lot of copy-pasting of markup code, and a range of other features. I'm more than happy to help with creating templates and dealing with markup. The MediaWiki instance hosted for us by VINCI is currently having authentication problems, however.

Timeline and goals

Already built: A database space within CDW to store lists of adjudicated elements. A SharePoint space for the phenomics core to store some documentation. A wiki system to store further documentation, allow for search, allow collaborative editing with easy versioning, and to help with SharePoint navigation. A very basic system to store source code (this system happens to be the wiki). Note that SharePoint has a wiki function too; also the SharePoint search function may be improved somewhat in summer, but I'm not sure to what extent.

Very short term (1–2 months): Decide on continuing with current wiki vs. SharePoint vs. a combination vs. some other solution. Organize the browsing function of the library to the specs of the phenomics core (Anne & I discussed this in the past).

Long term (1 year +): Agree on—and possibly train on—a more robust source code control system (or failing that, design the library to do the best we can). Agree on best practices for elements needed to document a phenotype.[4] Document at least one phenotype using these practices, to serve as an exemplar for others. Develop a system to advertise what is *in the process of being built* (not just completed tasks).

[4] E.g. who built this, when, using what process, for what chief purpose/study/intent, why did they make the choices they made, etc.

Even longer term: Have a system that infers a SME's internal mental model,[5] from the SME answering "keep/discard" to CDW elements, with only slight further effort from the SME.[6]

[5] E.g. "serum sodium is where LabChemTestName matches these 3 given text strings, where Topography matches these 2 given strings, or where LOINC is one of these 5 given codes, except in these 20 given cases which have probably erroneous LOINCs or TestNames. . . ."

[6] Nate Fillmore (BD-STEP fellow) has some ideas about how to approach this, e.g. a system that learns classification trees that the SME can select/edit at a high level.

Questions for the group

Do you agree with my list of requirements? Does a wiki fit the bill (see "Very short term" goals above)? Is the wiki the right place for SQL code? What training or further information on the wiki do you want? What should be captured about each phenotype?