# High Level

Recognizing the value of the NHLBI data to science, it is both bureaucratically and physically cumbersome to get access to. You have to get permission, and it can be large enough to be troublesome to transfer and process. Building a cloud utility can solve all three problems. Access agreements could be (have been) worked out so you sign up with the utility (PMP) and that carries over to the data providers allowing access to many data sources with just signing up once. Being able to process the data where it resides in the cloud means you don’t have to choke a DSL line or fill a disk with large files. Of course, a cloud can also provide access to clusters for heavy processing.

Harmonization comes into the picture for both dataset search and research. On the search side, a researcher would want to find relevant datasets by a number of criteria including a statistical description of the population studied. What data variables are available about each subject in the study and what is the range of values for each? This is in addition to obvious questions involving what medical condition was studied, what the inclusion criteria were, and what interventions or drugs were involved. Unfortunately, there is a Pandoras Box of names and values of attributes in the data from study to study, making search a challenge. For example, if one study calls sex “sex” and another calls it “gender”, a simple search on “sex” will miss studies that specify “gender”. The same problem appears in values if male and female are entered as 1 and 2 in one study, “male” and “female”, and “m”, and “f” in a second and third. The same problem extends to studying a combined dataset.

The high-level of the context extends beyond gaining access to the data and molding it to a suitable form to actually analyzing the data. This is often done using R or Python or their notebook forms with more elaborate presentation capabilities: Rmd from RStudio, or ipynb with IPython or Jupyter.

# The REAN Solution

I believe REANCloud customized a prototype designed and built by Bob Strahan and others at AWS called Harmonize, Search and Analyze. <https://aws.amazon.com/blogs/big-data/harmonize-search-and-analyze-loosely-coupled-datasets-on-aws/> The basic idea is to write Python scripts that read CSV files that convert data and add it to a Lucene index. Kibana makes for a pretty front end to Lucene. Look at the graphs here: <https://www.elastic.co/products/kibana> Kibana allows a researcher to find studies by a number of attributes. The data is then made available in a private area in the PMP where a researcher writes one-off R scripts to do analysis for papers.

It’s a great prototype and establishes a more clear baseline from which to continue development. The ways it falls short are points to focus future development on. The ways it succeeds are things to maintain. REAN’s work goes beyond the brief description above to address security concerns, cloud infrastructure and other things beyond my perception that are import to any solution going forward.

The chief shortcoming of this solution from a lab’s point of view is that the harmonization applied to the search index is isn’t reflected in the data supplied to the researcher after the search. Once a study is discovered as germane to a research effort, the data made available to the researcher is in its raw form as from the original source. Having the data in a harmonized form not only makes it feasible to combine datasets, but means code applied to one study can be more easily applied to another.

(2018-08-15) Arguably, the transparency of the mapping is a bigger issue.

However, there’s a growing awareness in the data world that reality is only ever relative. You could argue this started with the development of relational databases in the 80s. The problem with earlier network databases was hard-coded relationships, so the newer relational DBs allowed users to create whatever relationships they needed: the join. More recently this reality is reflected in a frustration with structure in data warehouses. The reaction is to build data lakes which are basically a capitulation to a world of different realities. Instead of trying to find a perfect single and universal structure for the data warehouse, just leave the data as-is and let the individual analysts do their ETL as they choose at the time of analysis. You can move the discussion from the context of corporate IT to Science if you change the language above from analyst and ETL to scientist and harmonization. The next conceptual step is to tie it to the Commons movement and FAIR principles with the realization that data harmonization is a big part of Interoperable data. Despite the futility of a universal standard, there is value in local standards within smaller, more realistic, more practical spheres of cooperation.

# Interlude: Glossary

* **Harmonization** Finding a set of common names, value units and specification granularity for data from more than one study to share so they can form a coherent whole. Statistics issues surrounding a coherent population without sampling bias are not addressed.
* **Categorization** Some analysis methods require variables with small numbers of discrete values rather than values from a continuous range. Creating age buckets like x < 40, 40 <= x < 60, x >= 60 for young, middle-aged, and old is an example of what we mean here by categorization.
* **Analysis Project, Analysis Matrix**. Once a study’s data is available in harmonized form and a cohort extract and categorized, it takes the form of a CSV file for analysis. This file is the output of a store of studies and the input to R or Python scripts. It would be the work by a scientist outside of the group developing the PMP, and outside the groups developing the studies’ harmonizations.
* **Rmd** a kind of R file used in RStudio that produces a PDF or HTML document.
* **Ipynb** similar to Rmd, but for Python using IPython. IPython NoteBook.
* **Git** a source control system. It keeps a series of revisions for a collection of files and allows users to add revisions in cooperation with each other.
* **Branch** a separate series of revisions from other branches or the main, master branch.
* **Merge** bringing tow branches together to merge the work of two or more people.
* **Workflow** a scheme and process of branches and merges to produce releases over a repeating development cycle. “gitflow” is one such workflow.

# The Kao Solution v1

Meanwhile, Dave has been working on more than one heart failure study and running into similar issues. He developed SQL code that harmonizes data so the same R script can be applied to many studies. At least a few papers have been published from this code using up to half a dozen studies.

# The Kao Solution v2: ADAPT-HF

ADAPT-HF involves moving earlier work to the PMP in a more general way. Recognizing the value of Ontologies in precisely defining a variable came from this work. Even before you consider crowd-sourcing harmonization is the problem of making a combined dataset. The question of how the harmonization was performed and to what standards comes up. Controlled vocabularies make it easier to define what the variables in the harmonized data set are. The definition guides conversion development and enables precise communication.

Deferring crowd-sourcing or shared community development of mapping development, the first step is to adopt the OHDSI OMOP CDM, a relational database schema for representing EHR data with references to controlled vocabularies to define the variables. Considering that the harmonization mappings will be shared as part of publications, this effort explores declarative approaches to the mapping, eschewing (often SQL) code-based approaches common in ETL. The mapping is a table listing the studies, files and variables and associating them with vocabulary terms and any necessary transformation functions and associated value mappings. Within limited scale, study search and cohort selection are SQL queries or applications of existing OHDSI applications. Early work maps a study id into the subject ids and selects all participants of one or more studies. “Ring Graph” like queries to find studies hasn’t been attempted.

# Enter Musen & Graybeal: CEDAR

The discussion above neglects search.

The REAN approach considers variable names and values as index elements as used in Lucene/Elastic Search.

Early thought on search in the OHDSI world is similar, though implemented in SQL on a relational database instead of Lucene.

CEDAR considers, to my understanding, not just the variable names and values, but higher level details, study level details, that describe the study without regard to a specific subject or patient. The condition studied, the intervention or drug used, the subject inclusion criteria, results etc.

From a software perspective, I assume it’s a database of sorts with a web app that could be deployed to AWS and provide links to study data packages much as Elastic Search does.

# Serious Scale of Harmonization Development

Sharing development of a mapping for a single study requires the ability to store more than one mapping for each study/file/variable so that each user can have their own version. There are two issues at play. One is about representing two differing ideas of what the ultimate mapping should be. The other questions the idea of a single ultimate mapping and allows for different views of reality. If more than one mapping is allowed per study, we need to consider how to deal within the CDM: multiple instances or a schema modified with a more complex primary key, perhaps not storing the data at all, just converting through as-needed.

# Harmonization Development vs Analysis Development vs Platform Development: Who Develops What?

In my mind, they are three different things with three different scopes of development communities.

The analysis projects are developed by a wide variety of people in relative isolation and take the form of a small R project. Each could be managed in a git repository with contributions managed by the individual owners and pull requests in a fashion similar to many open source projects. The maintainer doesn’t always have to be the original author and there is no restriction against forks. A directory listing of exports from git, or a more elaborate search page (with metadata of its own)

The Platform itself, because of security concerns relating to the source of the data should be kept under relative wraps. Not to hide the source, but to maintain control. Pull requests notwithstanding.

Harmonization seems to be the place with the greatest potential for community wide sharing of effort.