Large Scale Lookups, from an End-User and a Programmer Perspective

Peter Stegehuis and Naxin Zheng, Westat

Presenter: Peter Stegehuis

1. Introduction

With many lookups in a CAPI instrument, the contents of the lookup database are typically the same for every survey respondent. When we ask respondents what prescribed medicines they use, for example, the list we use in the background for their search is the same for all respondents. However, when we want them to select a physician or medical professional in a lookup search, it doesn't make sense to present all possible selections from the entire country to everyone, as the lookup would be too large and, even more importantly, it would end up showing many irrelevant results to the end user, making it too hard to find the desired result. This paper discusses ways to make medical provider lookups relevant from an end-user perspective and manageable from a programming and file management viewpoint.

2. The Challenge

We are asking respondents for medical information, including doctor and hospital visits, over the period since the previous interview in the panel survey. Even with the help of records at hand, it can be difficult for respondents to recall this information for all household members, especially if the reference period is, for instance, six months long. There is also a separate follow-up survey with the providers, to get more accurate data on the exact procedures and cost. In order for providers to be able disclose this data, we ask household members to sign so-called forms to authorize data collection from providers. It is therefore crucial to collect accurate data on the providers and their contact information.

3. What is Needed?

So, rather than letting the interviewer just type provider names, addresses, and phone numbers as remembered by the respondent, we want to have them select a record from a lookup that has accurate medical provider information.

In order to implement this within the interview program, we need to have the following elements in place:

- A good resource for provider records
- A mechanism for a good lookup
- Meaningful, localized lookup data files
- A user-friendly presentation of lookup search results

We will describe these elements in the following sections.

4. Resource for Provider Data Records

There are different data sets of medical providers in the entire US available, some publicly and some for an annual fee. We have looked at a number of them and found that the most comprehensive data set seems to be the National Provider Index (NPI).

There is a strong incentive for medical providers to sign up and get a unique NPI ID: any provider—person or facility—who wants to ever bill a federal entity like Medicare or Medicaid needs to have an

NPI ID. This makes the NPI a more comprehensive list of providers than others, a crucial distinction for our purpose.

It is far from a perfect list however: there is no incentive for providers to delist when a doctor retires or a medical facility shuts down or moves. This means that the list keeps growing year over year, and many entries are no longer relevant. It also means that the lookup search mechanism becomes of even greater importance if the interviewer is to find the correct medical provider in an enormous mix of useful and useless records.

5. The Search Mechanism

To be actually useful during an already long interview with possibly many provider searches, any search mechanism to be used for this search has to be fast and powerful, serving up search results almost instantaneously after an interviewer types in a search string. It also needs to be somewhat forgiving of typos in name or address of a provider as given by the respondent and allow for easy searches on name, phone number, and/or address.

Before we switched to Blaise 4.8, we did this search in SQL Server, using a "LIKE" search with wildcards. The speed of a search was adequate, but that mechanism was not able to deal with typos very well, especially if they were at the start of search words. It also needed separate searches for name, address, and phone number.

The switch to Blaise 4.8 and trigram lookups meant a big improvement in the success rate of provider searches. Because of the trigrams—three letter snippets—this search mechanism is much less influenced by a typo, and the way Blaise indexes those trigrams means the search results come up extremely fast. With a carefully constructed 'search string' for each record in the lookup data file, we can also assure that the interviewer can choose to enter parts of the provider name, address, phone number, or some combination of these without having to specify that in any way by designating a search category beforehand. We have seen much better outcomes since we switched, with a much higher percentage of searches finding a match, as opposed to adding verbatim provider info after an unsuccessful search.

5.1 Creating Localized Lookup Data Files

One way to try to include relevant providers for a specific interview is to use the respondent's state: include all providers within that state and exclude all others. The problem with that approach is that people will cross state lines to get medical services, and for many people near state lines, that may well be their preferred option. The same disadvantage goes for the counties within states.

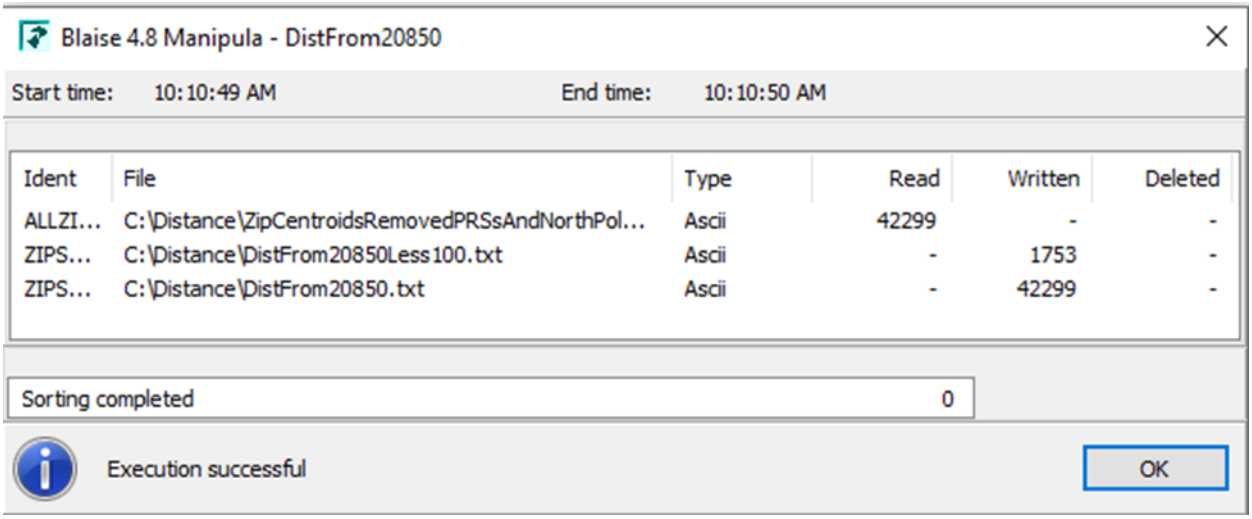
Instead, we wanted to ensure that the respondent's address was more in the center of the area that would be covered by the lookup file. So, we started off with the respondent's zip code (for non-US readers, that's a postal code, much smaller than states or counties). Just including providers within that zip code would be too limiting, so we wanted to include all the providers within a wider circle around the respondent's zip code.

At the outset, we created circles with a radius of 100 miles around the zip code's centroid (you can think of that as the center or average location in the zip code area). The list of zip codes in the US, and their centroid coordinates, is freely available. So, for each zip code, we calculated a list of zip codes that would fit within the circle with radius of 100 miles by calculating the distance from one centroid to another.

As an aside, Manipula is known for being very fast, especially in handling large text files. We had to double check the results when calculating the distance from zip code 20850 to all others—writing two

output files and sorting the first output file took all of one second. Figure 1 is a screenshot with that result, with start time and end time near the top.

Figure 1. 42,000 Distance Calculations in One Second



Unfortunately for us, creating Blaise trigram lookup files takes a lot longer—not surprising when thinking about the work needed for creating, indexing, and managing the trigram index file.

The next issue was when and where to create these lookup files, and how to get them on interviewer laptops. There are more than 42,000 zip codes in the US and creating and storing them all takes up 20 TB (or 2 TB zipped), which is, of course, way too much data for storing on an interviewer's laptop. We can't create them during the interview either, as it can take up to 15 or 20 minutes—and lots of resources—to create some of the bigger lookup files. So, we are creating them on the interviewer laptops: right after the interviewer has picked up the transmission with the preload data for that case, the IMS checks whether lookup files for zip code and state are already on the laptop. If not, they get created at that point.

We tried to optimize the Manipula process that creates the lookups to minimize the time needed when interviewers get a big number of cases assigned at once. Somewhat to our surprise, we found that splitting the process in two separate Manipula setups was faster than using just one setup.

The annually growing file with NPI records is too big—close to seven million records at the moment—to fit into memory in a Manipula setup, where that would have been useful. So, using a two-step process, each one using a differently sorted file, we ended up with a combined process that was much faster than the one-step approach. This was much appreciated by field staff and home office staff who sometimes had to wait for the completion of this task.

We have also reduced the size of some of the biggest lookup files by basing the diameter of the circle around the zip code centroid on the urbanicity degree of the area (this is also freely available data). The thinking there is that there are many more medical providers in urban areas compared to rural areas, so respondents would typically not travel as far to see a provider. This means that instead of a 100-mile diameter, we can have a much smaller diameter in, say, New York City compared to a rural area in, for instance, Wyoming.

On the other hand, the process gets slightly more complicated by choosing to include certain providers in all lookup files, regardless of their zip code. These are what we call 'Centers of Excellence,' with

examples like the Mayo Clinic, Sloan Kettering, Johns Hopkins, and other well-known medical facilities. People may travel to these centers for consultation and/or treatment, including surgery. While these cases may be relatively rare, they may include extensive and expensive medical treatment. Including these facilities in every lookup file helps us to capture everything about these visits as accurately as possible.

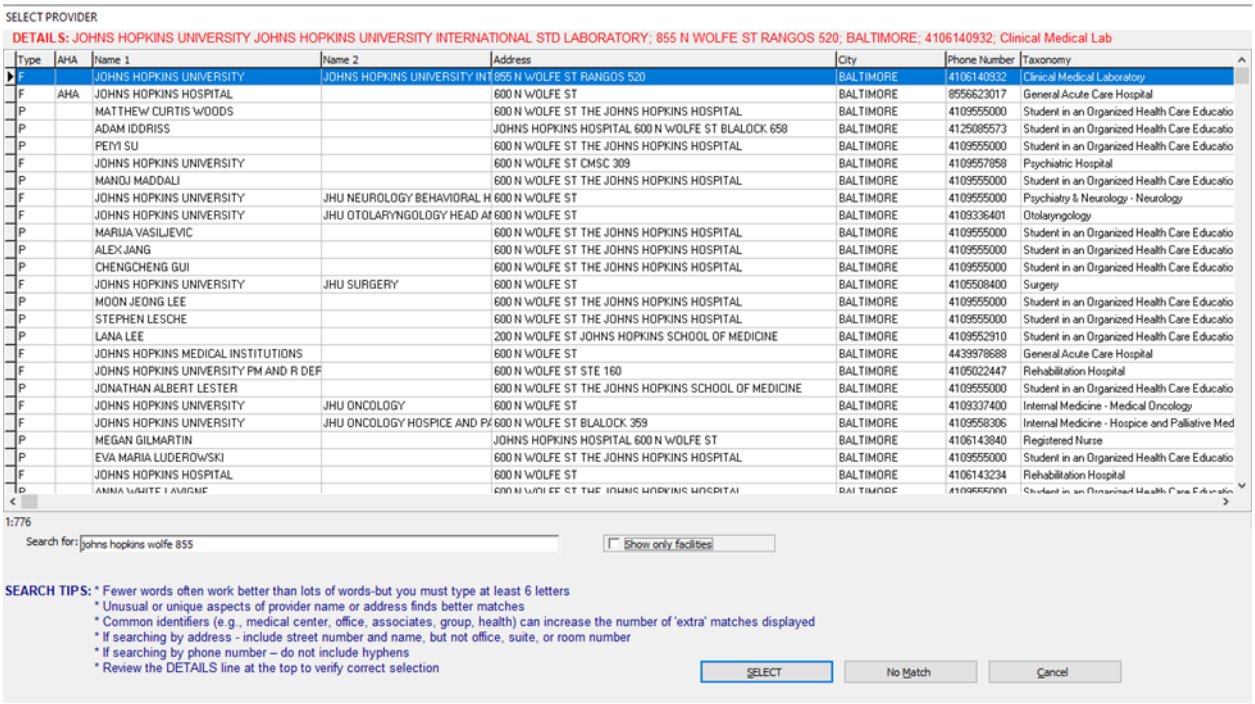
6. Presentation

Lookups, including trigram lookups, have been available in Blaise for decades, so Blaise users are likely to be very familiar with the look and feel of these screens. There are, however, some elements that can be tailored or added to make a lookup more user-friendly and more effective, as well.

We are using a Manipula dialog for the display of the lookup, which allows us to add a few elements that are useful to interviewers.

Figure 2 is a screenshot of a typical search.

Figure 2. Medical Provider Lookup Screen

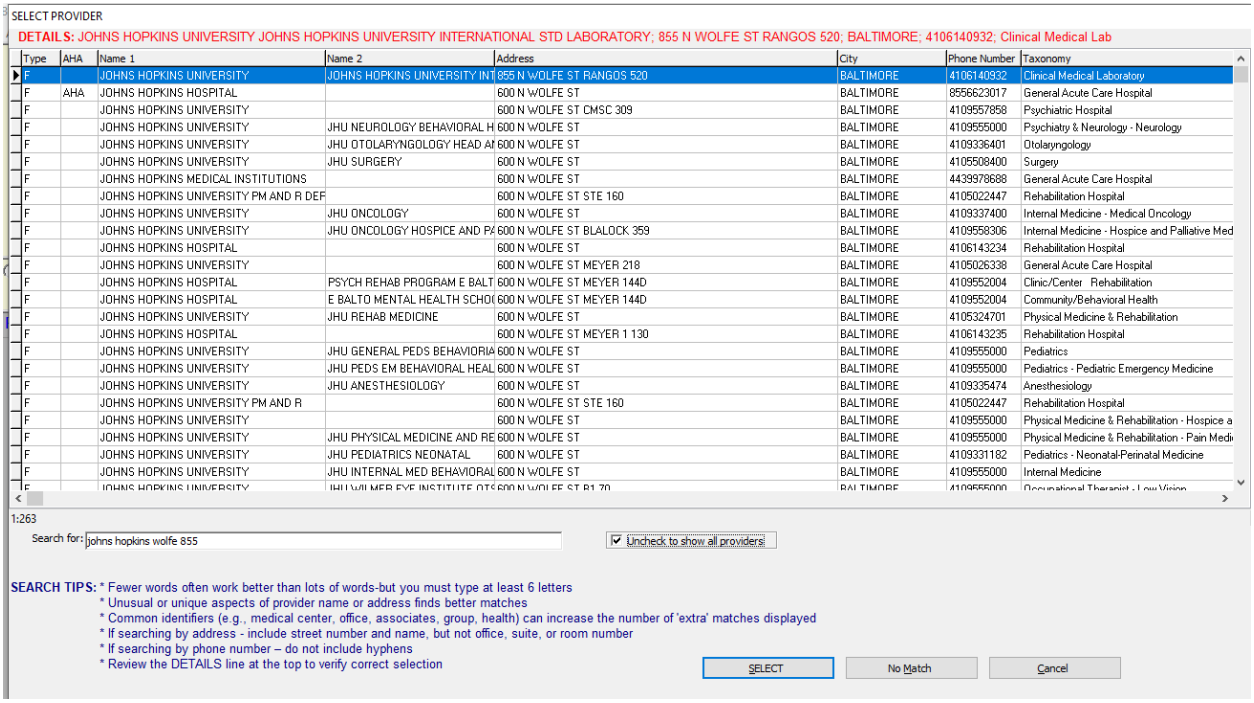


Note the red line at the top, which shows the detailed information of the currently selected line in the results grid. This ‘detail line’ gets updated automatically whenever a different line is selected and it allows the interviewer to easily see all the available info on that provider, even parts that might be cut off in the results grid.

There are search tips for interviewers near the bottom because it can really be a difficult task to help a respondent recall details of an event and names of doctors and facilities, even more so if the event happened several months ago.

We also included an option to filter the results and display only facilities, instead of doctors and facilities all mixed together. As you can see in Figure 3, we have a checkbox with the label “Show only facilities” next to the searchstring input line. When checked, we get the following:

Figure 3. Medical Provider Lookup Screen with a Filter to Show Only Medical Facilities



Note that all records now are facilities and that the label for the checkbox now reads “Uncheck to show all providers.”

This capability to filter search results is quite naturally built-in in Blaise 5, but in Blaise 4.8, it is a fairly well-hidden feature. Using it requires a decent amount of work, and if you want to do your own sorting of search results, it also requires a recalculation of trigram scores, as those are not passed on with the search results.

The searchstring that the interviewer types to try to select a provider can be accessed and stored for later analysis. When no match can be found in the lookup, the interviewer can select the “No Match” button and enter provider details verbatim.

7. Conclusions

Trigram lookups in Blaise work very well and fast, even with very large lookup files. This is as true in Blaise 5 as well as it was in Blaise 4. In Blaise 5, the additional flexibility for presentation and the ease of filtering search results are much appreciated.