Data Storage and Organization

Alan Whitney

This talk will address the basic organization of a nutrient data base using examples from the HVH-CWRU Nutrient Data Base. Techniques for increasing the flexibility of the data base and application programs will also be discussed.

The primary thing one is interested in when using a nutrient data base is a set of nutrient values for a set of food items. The data base can therefore be modeled as an array of nutrient values. There are other things that are useful in the data file. The first is the food item identifier and a description of the item. It is useful to include encoded quantity information to allow scaling of values in the data base to other convenient units. For example, nutrient values in the data base may be for some standard amount of food, like a 100 gram portion. It is more convenient for the user to access the data in a more usual measurement such as a volumetric or descriptive measure. Additionally, several other data items can be used for documentation of sources and revisions and other coded information relating to the food item. This leads to the storage structure shown in Figure 1.

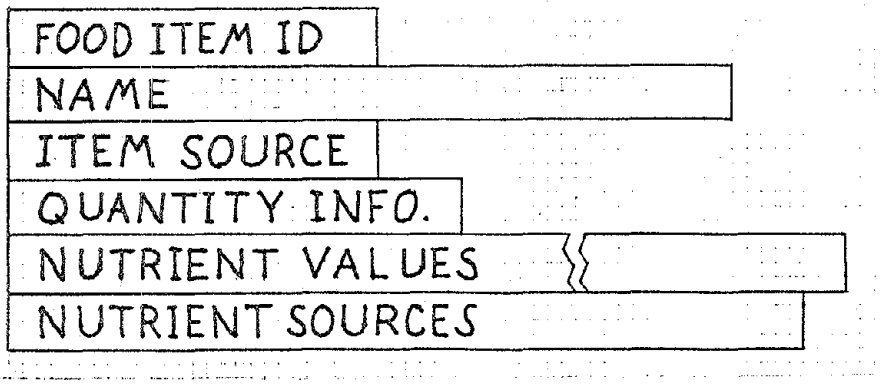

Figure 1.

Structure of a Food Table Record.

If the key into the data base, the food item identifier, is a number in the range one to the number of items in the data

base, this number can be used as an index into the data file. If the item identifier is more complicated, it is convenient to have a separate index file that gives the identifier and the index into the data file for each item. Note that space can be saved in the index file if the data file is stored sequentially. In this case, the index is just the position in the index file of the index record.

A partially implemented feature of our system is a distinct recipe file which describes a recipe item as a combination of food items. This file is structured similarly to the nutrient data file except that instead of having the stored nutrient data, there is a list of food item identifiers and a factor which represents the proportion of that item in the recipe item. This organization does not allow for different cooking losses for different nutrients, but it does insure that the most current nutrient data are used.

A drawback to the data organization described is that it does not take advantage of the fact that many unknown nutrient values are present in the data base. Currently, unknown values and trace values are represented by an impossible data value. It would be possible to save storage space by including with each data value a nutrient code which indicates the nutrient represented by this particular value. Unknown values would then be represented by the absence of a value. One advantage to this organization is that the set of nutrients to be considered could easily be expanded. A drawback is that this would necessitate using variable-length records or multiple fixed-length records and would complicate the access to the data. When a record of this form is read, it could be translated to a simpler form.

Another desirable goal would be to parameterize the structure of the data records so that a single program could access multiple data bases in different formats without modification. This could be implemented using an additional file that describes the location and size of the data fields of interest. For example, Figure 2 might be a description of the HVH-CWRU Nutrient Data Base. The first line of the description gives the name of the nutrient data file. The next line gives the size in bytes of each record in the file. The third line gives the location, size and format of the item identifier. In this example it is indicated to be at byte offset 0 and consist of two binary integers. The fourth line gives the location and length of the descriptive information for the item. The fifth line gives the number of nutrient values in the data record and the remaining lines describe each nutrient data field. A nutrient data field can be described as the position and format of the data and a set of descriptive information that can be printed on output reports. The descriptive information given in the example is the nutrient name, an optional shortened form of the name and an abbreviation for the units nutrient values are reported in. There are tradeoffs in this representation. Quantity codes are difficult to represent and assumptions about their specific interpretation may

```
/usr/hvh/data/ftab.n
464
0 2bi
4 60c
71
98:Water:Gm
102:Kilocalories,KCal:
106:Total Protein,T-Pro:Gm
110:Animal Protein,A-Pro:Gm
114:Plant Protein,P-Pro:Gm
118:Total Fat,T-Fat:Gm
122:Animal Fat,A-Fat:Gm
126:Plant Fat,P-Fat:Gm
130:Total Carbohydrate,T-CHO:Gm

        . . .

370:Selenium,Selen:mg
374:Sulphur,Sulph:mg
378:Zinc:mg
```

Figure 2.

Sample Data Description File.

have to be made. An interpreter of the description file must be added to the application programs. Added complexity in the data file structure would lead to more complexity in the file description and the interpreter. The compressed data file structure discussed previously would be very difficult to describe concisely.

Increased flexibility could be achieved using a general purpose data base management system (DBMS). These systems usually allow data file formats to be redefined without much difficulty. The tradeoff in this approach is the additional storage that will be required and the increased access time. Use of a DBMS may also limit the choice of an application programming language.

Another area where a parameterization approach might prove useful is for computing nutrient standards. It would be straightforward to set up a nutrient standard file that describes a nutrient standard as a percentage of some constant value. Figure 3 shows a description of the percentage of U.S. RDA for the HVH-CWRU Nutrient Data Base. The first line gives the title of the standard and indicates that it is to be calculated as a percentage. Succeeding lines then give the nutrient number to be considered and the constant value to be compared with. The last

```
%USRDA
3  65      Protein
47 5000    Vitamin A
37 60      Vitamin C
38 1.5     Thiamine
40 1.7     Riboflavin
39 20      Niacin
57 1000    Ca
56 18      Fe
53 400     Vitamin D
50 30      Vitamin E
41 2000    Vitamin B6
43 .4      Folic Acid
42 6       Vitamin B12
58 1000    P
61 .15     I
62 400     Mg
71 15      Zn
66 2       Cu
44 300     Biotin
46 10      Pantothenic Acid
```

Figure 3.

Sample Nutrient Standard File.

item on the line indicates the nutrient under consideration. This representation works well in this example, but would not be applicable to RDA's based on functions of sex, age, weight and other nutrient values. Consider the RDA for niacin as defined by the text of the 1974 RDA. This standard is a function of age, sex, niacin intake and other nutrient intakes. It would be possible to define a representation that would work, but the complexity of creating the files and the interpreter for them would be much greater than hard-coding the formulas.

Questions and Answers:

Q: What application programming languages do you use?

A: Our primary programming language is Fortran and we do some programming in a language called C which is a more modern language.

Q: What percentage of fields in your data base are unfilled?

A: It really depends on the field. There are some fields for which we have values for almost everything, every item that is. Examples are calories, total fat, total carbohydrate and total protein. There are others where there's almost no information. Examples, iodine, I believe there's not a whole lot of B6 information, and things like that. Some of these are nutrients which are in the U.S. RDA and the RDA's and I really can't come up with an estimate of altogether how many fields are vacant or unknown, but we have been thinking about actually quantitifying that.

Q: How many structures do you provide and how do you relate them to the available government tapes?

A: The linkage between the item identifier and the government tapes is through the field in the item record which is the item source. Item source is encoded to be the same number as the government tape numbers or the publication numbers and there's also some alphabetic information that indicates whether it's publication or food manufacturer's data or from USDA. Food item ID is structured as a six character numeric ID the first two characters of which are a food group identifier. And that's represented in the actual file as two binary integers rather than as character information.

Q: What are the other four characters?

A: The other four characters are just a number. We do separate some food groups into food subgroups, so the first character or two of that can indicate that. But otherwise it's just some unique identifying number. Usually with space between sequential items.

Q: I'd just like to ask about your food groups. You've indicated the first two characters were food group identifiers. I think Ohio State is using a four character identifier which brought o t a more detailed breakdown of foods. What system do you provide?

A: I don't know if I'm really qualified to answer that question. I really didn't have anything to do with constructing the food groupings and things like that but we have found that some don't work real well. There are certain things that are difficult to classify.

Q: What about items that have multiple groups that they might be a part of?

A: That's the basic problem. How to classify those. It's difficult to come up with some consistent way of doing it.

Q: Do you make any attempt to assign a scientific name if it's a fruit or a vegetable?

A: No, not really, however through the ID number, if it's a USDA item I believe that you can go back to that information and find out scientific names. But that's not normally part of our descriptive information.

Q: Alan, I'D like to suggest for foods that are cross-referenced, for example, catsup you might think is that a condiment or is it a tomato product? In the coding manual, under tomato, you may say, "Catsup, see condiments".

A: That's right, I forgot about that. We put together a special coding manual to go with the system and it sometimes indicates cross-references so that from one section of the manual it might be referenced to some other section.

Q: But that means you're imposing kind of an arbitrary structure on your coding and that's one of the logical dilemmas that we have in making the data useful to the user.

A: I think it would be difficult to come up with some convenient, consistent system of doing that. I don't know though.

Q: I was wondering about your nutrient sources code. Do you have identified in that code the actual food item identifier that's identical to the one in Handbook 8 or USDA?

A: The item source identifier indicates whether the item was actually taken from USDA publications.

Q: In other words, you can't overlay your tape with USDA's.

A: Not really, not directly. If the item is identified as a USDA item, there are alpha-numeric codes that tell you which particular nutrients were taken from USDA sources. If additional data were taken from other places, that's indicated also.

Q: Is your source code printed out on hard copies or are these codes part of the record?

A: We don't typically print those out on the actual dietary analysis and things like that. If questions come up we can reference them and ...

Q: I'm speaking of the hard copy of the data base.

A: Yes, we have hard copies of the meanings of the codes and things like that.

Q: ... printed out with the hard copy?

A: No, you just print out the code itself and then you can reference another sheet to find out where it came from.

Q: What about the components of the last 4 digits of the identification?

A: They're just four numeric digits. Sometimes they indicate some subgrouping within a major food group.

Q: Then how many digits do you use to actually identify the different foods within the subgroups or groups?

A: We use all four of them. It's sort of a sparse identification. In other words, you can have spaces inbetween the numbers. For example, the first food item number in the table is 01- which indicates a beverage, and then 0010 which indicates some particular food item within that group. It doesn't give you any indication whether it's the first or last item, except that if it were 0001, then it would have to be first.

Q: When a record is processed, do you store that information for use later on?

A: No we don't. You mean, for example, a dietary or something like that, store the accumulated values or something like that? No, we don't do that, we consider that it isn't that difficult to re-run it if need be. Especially since you may want to make changes in the input or the output.

Q: If you wanted data say on 100 foods in your data bank, could you print it out in a matter of a few minutes?

A: I expect so.

Q: In terms of the nutritional values for them?

A: Yes.