

# SOPHIEs made easy: Part 4



**Dr Robert Trecharne-Jones**

GP, Torquay

This is the fourth part in the series designed to take users through the step-by-step approach to the Graphical SOPHIE Editor. This edition we will start to look at each of the SOPHIE 'questions' in some detail. This is what SOPHIE is all about—translating a protocol into its fundamental constituents—and it's just a pity that there is so much groundwork to be done before getting this far!

Let's just reiterate something we found out in Part 2, which is the way SOPHIE questions fall into four distinct types:

- **Database questions.** These require no input from the user; by using a database question in your guideline you will be asking the software to look into the electronic patient record and to retrieve some element of the patient's registration or clinical details.
- **Information questions.** This type of question is another which requires no input from the user. It's simply a point of information to the user, often used to forewarn the practitioner about what they can expect from the next section of the guideline.
- **Patient questions.** This is the type of question which does require input from the user, usually as a result of information gleaned from the patient during the history or examination.
- **Function questions.** This most recent addition to the SOPHIE engine allows the user to fire up another software application from within the SOPHIE guideline.

## \$SEX

We're now going back to the first question of the first section—the \$SEX question. (The reason for the dollar \$ sign is a simple convention to indicate that we are talking about a SOPHIE question rather than some other reason for writing SEX in capital letters!) When we came across this question in Part 2 of this series we used the question simply as an example to discuss the exit options, but now it's time to look at the uses for this question in real protocols.

It's easy to find \$SEX on the palette because that's exactly what it says on the label! Just left-click with your mouse on the Sex icon, and then left-click again with your mouse to place the cursor on the blank drawing board, and holding down the left button of

your mouse, drag the mouse down and to the right, to open up the question 'box'. If you have done it correctly you'll see something like this:-



What this question does is refer back to the patient record to determine the sex of the patient, and that gives the guideline designer the chance to jump over one or more questions which are unnecessary for patients of a particular gender. The crucial point to remember for this and any other database question is that *it requires no input from the user*, and this results in a protocol that runs more quickly and becomes more intuitive. It would be a complete waste of time, for instance, to include a Yes/ No option instead, asking the user to confirm that this patient was a man, for instance, when the whole thing can be done automatically.

An obvious example for this question relates to a new patient screening protocol in which you want to ask about contraception. By including a \$SEX question at the start of this section you can ensure that males are omitted from the next section, while female patients can be asked other questions relating specifically to matters of contraception.

You will find in practice that the \$SEX question is used far more often to include extra questions for women than for any other purpose, although in theory there is no reason why this should be the case.

The options on leaving the question are the same as those discussed in Part 3 of this series (*Notes/Medication/Advice/End*) and these options are exactly the same whether you select the male or female exit path. Obviously you can make use of these if you so wish, to add Read codes or prescribe medication, but in my experience these fields are usually left blank when the \$SEX question is used—the question is being used in its most fundamental form to split the protocol down different branches.

And now it's time to answer a quirky point I raised in a previous issue. At the bottom of the SEX question box are three small icons which denote, left to right, the path taken if the patient is male, the path taken if the patient is female, and the path taken whatever the gender of the patient. What is the point of including the question in the first place if the answer will be the same whatever the gender?

These next two paragraphs are

necessarily convoluted. They are not essential reading, but those who wish to know more about the subject might want to wallow their way through.

The answer to the question is that although it may not happen just yet, one day you will be designing quite large protocols, with complex branching structures. You won't get it right first time, and you will want to rearrange great chunks of the protocol to sort out the question order for greater ease of use. Imagine the confusion if you had to start deleting and redrawing the connecting lines to bring together loose fragments of different branches of the hierarchy! By using the \$SEX question at the start and finish of all the major sections within a protocol, and by making use of the 'either gender' option in each case, you can tie up all the loose ends and connecting lines in one section before moving on to the next. Then if you change your mind later on, and want to rearrange a couple of sections, you only have to change the connecting lines joining the sex questions at the end of one section and the beginning of another.

Why do I have to use the \$SEX question in this way, you might ask. Why can't I use any one of the other questions for the same sort of purpose? Well, you would have to use one of the other database questions to avoid the need for any user input, and none of the other database questions is quite so straightforward as the \$SEX question.

To those readers who struggled through, it's time to lighten up a bit, and to those who skipped the section completely, welcome back! It's time to move on to \$AGE.

## \$AGE

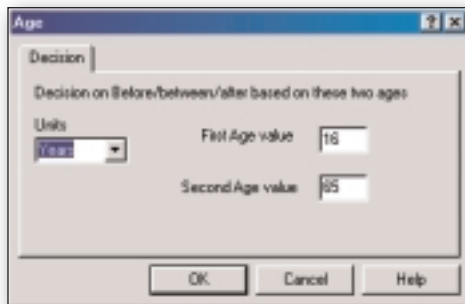
The \$AGE question is exactly analogous to the \$SEX question but this time it is examining the database to determine the patient's age and offering options accordingly. Open up the \$AGE question from the palette and let's see what we've got:-



The top line in the question is where we set the age parameters on which the (optional) decisions can be made later. You can see that the default age parameters are 16:65 years, which means that the three exit options have the following significance:

- The patient is 16 years old or less
- The patient is aged 17 to 64 inclusive
- The patient is aged 65 years or more

Of course you can set the age parameters to any margins you like. Double-click with your mouse in the little yellow clock at the top of the question, and this is what you get:-



Using the drop down menu arrow on the right of the Units box you can select 'Days, Weeks, Months or Years' while the other two fields relate to the upper and lower numeric limits you require for this particular question in the context in which you will use it.

You may imagine that 'Years' is the commonest unit that is used in practice, and the reason that the question defaults to 16:65 is because New Patient Screening, for example, frequently includes chunks of questions that are specific to children, adults and pensioners, and these limits form useful cut-off points.

**It is important to remember that the < option, for instance, does not mean 'Less Than'.** It actually signifies 'Equal

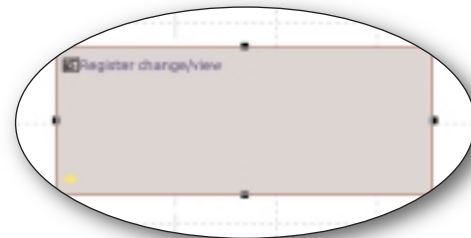
To or Less Than' so if you leave the question on its default setting this is the branch that will be taken if your patient is 16 years old and even if he is just a couple of days short of his seventeenth birthday! You will find from practical experience that there are some contexts where it is important to remember this point. The corollary is that **the > option implies 'Equal To or Greater Than'.**

What if you want to split your patients into more than three age bands? Easy: just put two or more \$AGE questions one after the other! For instance if you wanted to subdivide your older patients into those aged 65-74 and another group aged 75 and over it is a simple matter to lead the > option from the first \$AGE question into a second \$AGE question where the parameters are set up as, for instance 50:75. Note that it doesn't actually matter what the first figure is, because all patients getting this far will be at least 65 years old anyway, so the only exit options you need to configure are the > pathway, for those aged 75 and over, and the = pathway for those aged 65-74.

## \$REG

The Register question is currently only available in System 5 and it allows the user to duck out of the SOPHIE and go straight to the registration page (*see figure at the top of the next column*).

This begs the obvious question 'Why on earth would I want to do that?' to which I can offer an immediate and practical response. One of the first



SOPHIEs I ever wrote for my own practice allowed users to record death details. It offered prompts to complete the death certificate and if the patient was to be cremated it even searched the record to look for evidence of pacemakers or radio-active implants, and warned the clinician to remove these items. It then took the user into the registration page to allow them to change the category from '3' to 'D' if this had not already been done.

## Setting the target system

So how does the Graphical SOPHIE Editor know whether you are using System 5 or System 6000? After all the end result is the same—a GDL file that you can plug straight into your system. The Editor requires you to designate your 'target system' by using the *View/Options* menu in which there is a 'Target' tab. Click on System 6000 and the Register icon on the palette becomes greyed out, only to reappear if you click on System 5 again.

*Next time we'll move on down the SOPHIE questions including the powerful options where the user has a part to play!*