Subject: Is there a standard 'null' array?
Posted by Matt Feinstein on Fri, 28 Mar 2003 13:47:01 GMT
View Forum Message <> Reply to Message

This is an embarassingly elementary question, and I'm -sure- that it has a simple, elegant, and obvious answer, but...

I often find myself wanting to do things kinda like

```
Data_I_want_vec =  Init
...
for ix = 0 , (size(data_array))[1]-1 do    $
  if condition(data_array[ix]) then          $
    Data_I_want_vec = [Data_I_want_vec, data_array[ix]]
```

I realize that there is analogous code for a vectorized version of my question, but I think the for-loop version is clearer.

The questions that arise are things like: What to use for 'Init' ? What do I do to test for a null result? What if there's no data?

Some notes:
a) Matlab has an empty vector = [] that serves these purposes. It's possible that my desire to program in this fashion is simply a Matlab-ism, and I need instruction in the IDL way.
b) I do not, really and truly, need instruction in the use of the WHERE function or in the use of vectorization. Believe me, I know all about that. The analog of my question in the vectorized case is that it looks like I have to surround the vectorized expression with boring and error-prone tests for null inputs and outputs.

Is there a 'standard' way of doing this sort of thing?

Matt Feinstein

--
The Law of Polarity: The probability of wiring a battery with the correct polarity is (1/2)^N, where N is the number of times you try to connect it.