September, 2000

Advisor Answers

Using System Menu Items

Visual FoxPro 5.0/6.0

Q: I would like to programmatically call a VFP System Menu Bar, in this case, _mfi_export. I'm trying to use this functionality without having a system menu at all.

The best I can discover is that you can set up a CTRL key combination (provided a macro does not override it), and you need only to have defined the menu, not necessarily activated it.

Is there any cleaner/simpler way to call this function from, for example, a Command Button?

–Ian McColm (via Advisor.COM)

A: In fact, beginning in VFP 5.0, there is a better way. The SYS(1500) command allows you to call a system menu bar directly without having to resort to any tricks like KEYBOARDing a menu shortcut.

As you learned, the menu item in question must be defined, but does not have to be activated. So, to make the Export item available, you can do something like:

```
DEFINE POPUP _MFILE
DEFINE BAR _MFI_EXPORT OF _MFILE PROMPT "Export"
```

Then, your Export button can use this menu item by calling:

```
SYS(1500, "_MFI_EXPORT", "_MFILE")
```

Interestingly, the menu bar doesn't have to reside on the same menu popup where it usually lives. You can put it wherever you want as long as it has the appropriate name. However, the menu popup does have to have the name of a system menu popup; you can't just assign any old name. Putting the menu bar on a different menu popup is particularly handy in two situations.

First, you may want to make a whole bunch of menu items available without having a system menu. There's no reason to define multiple menu popups. Put them all on a single popup that uses the name of one of the system menu popups. Just be sure that your calls to

SYS(1500) use the appropriate popup name. For example, you might issue these definitions:

```
DEFINE POPUP _MEDIT
DEFINE BAR _MFI_EXPORT OF _MEDIT PROMPT "Export"
DEFINE BAR _MED_CUT OF _MEDIT PROMPT "Cut"
DEFINE BAR _MED_COPY OF _MEDIT PROMPT "Copy"
DEFINE BAR _MED_PASTE OF _MEDIT PROMPT "Paste"
```

In that case, to call the Export item, you'd use this call:

```
SYS(1500, "_MFI_EXPORT", "_MEDIT")
```

Second, there may be times when you want to use a system menu item in this way, but the popup to which it belongs is defined as part of a visible system menu. In that case, you can put the menu bars on one of the system menu popups that can't be used in a distributed application, such as _MDATA or _MRECORD. Again, in this situation, be sure to name the appropriate popup in your call to SYS(1500).

When you decide which menu items to make available this way, keep in mind that some items on the VFP menu cannot be distributed with compiled applications. Unfortunately, Export is such an item. It's clear from the start that Import is trickier because the dialog it brings up includes a button for the Import Wizard and the VFP wizards cannot be distributed. However, testing shows that Export also doesn't work in an .EXE. There are many other items in the menu system that cannot be included in a compiled application, as well. (See the topic "Removing Restricted Visual FoxPro Features and Files" in Chapter 25 of the Programmer's Guide in the VFP 6.0 Help file for a list of items that cannot be distributed.)

A number of other items need the right conditions in order to work. For example, Cut, Copy and Paste only work in an editing context. I wrote an article that appeared in the November '97 issue of FoxPro Advisor that showed how to create an editing toolbar that keeps itself properly enabled and disabled, using SYS(1500) and a timer.

–Tamar