

**#24            Speeding the filling of scrollable areas**

Written by Dave Terry

Published Date December 14, 1987

Rapid methods of filling 4th Dimension's arrays.

Scrollable areas are often filled on a per record or per subrecord basis. In both cases it is common to use a While loop or an Apply To Selection/Subselection command to step through the records/subrecords and fill the array. Usually the code would look something like this:

**Using While**

```
SORT SUBSELECTION([filename]Sub;[filename]Sub'Subfield;>)
Array0:=0
While (Not(End subselection([filename]Sub))
      Array0:=Array0+1
      Array{Array0}:=[filename]fieldname
      NEXT SUBRECORD([filename]Sub)
End while
```

**Using Apply**

```
SORT SUBSELECTION([filename]Sub;[filename]Sub'Subfield;>)
Array0:=Records in subselection([filename]Sub)
i:=0
APPLY TO SUBSELECTION([filename]Sub;Fill SubArray)
```

**Procedure Fill SubArray**

```
i:=i+1
Array{i}:=[filename]Sub'Subfield
```

In the Apply example, i is used as the counter in place of Array0. If Array0 is used the scrollable area will not update properly. The Apply To Selection/Subselection command is more elegant and usually a bit faster.

**A faster approach** to filling an array is to have an integer field in the file/subfile that is sequentially numbered (in this example "Seq" is the subfield name). With this approach the array can be filled in one line of code.

```
SORT SUBSELECTION([filename]Sub;[filename]Sub'Seq;>)
Array0:=Records in subselection([filename]Sub)
APPLY TO SUBSELECTION([filename]Sub;Array{[filename]Sub'Seq}:=[filename]Sub'Subfield)
```





This method is suggested here for subrecords. In order to keep the subrecords sequential, the subrecords must be renumbered after every deletion or insertion. This is a fast process because all of the subrecords for a particular record are in memory at the same time and thus are easily renumbered. However, keeping a large number of records sequential could be time consuming because after every deletion or insertion many records might need to be altered. This approach could be used to fill arrays with records if the information is static (no deletions) and the array is always needed in the same order.

