Rocket software

Rocket Uniface Library 10.4

Example: Moving Structs

This example demonstrates how an existing Struct can be inserted as a member of another Struct.

```
function MOVE_STRUCT
variables
    struct vStruct1, vStruct2, vSteadyRef
endvariables

  call printHeader("MOVE_STRUCT") ; display entry header in the message frame

  ; Build Struct vStruct1
  vStruct1->a = "AAA"
  vStruct1->c = "CCC"

  ; Build Struct vStruct2
  vStruct2->b = "BBB"

  ; Make vStruct2->b a member of vStruct1:
  vStruct2->b->$parent = vStruct1       [1]
  putmess "Struct vStruct2->b moved to vStruct1, and appended as member 'b':"
  putmess vStruct1->$dbgstring

  ; Reposition the newly-added Struct (assuming this is relevant)
  vStruct1->b->$index = 2               [2]
  putmess "Newly-added Struct 'b' has been repositioned:"
  putmess vStruct1->$dbgstring

  ; Alternative: Use a dedicated reference for manipulation
  vStruct2->b = "BBB"
  vSteadyRef = vStruct2->b    ; vSteadyRef refers to the 'b' Struct    [3]
  vSteadyRef->$parent = vStruct1
  vSteadyRef->$index = 2
end ; function MOVE_STRUCT
```

1. Move a Struct by changing its parent using the `$parent` Struct function. The moved Struct is appended to the other Struct.

   ```
   ================================
    MOVE_STRUCT
   ================================
   Struct vStruct2->b moved to vStruct1, and appended as member 'b':
   []
    [a] = "AAA"
    [c] = "CCC"
    [b] = "BBB"
   ```

2. Move a Struct member to another position in its parent Struct by assigning it to the desired index using the `$index` Struct function.

   ```
   Newly-added Struct 'b' has been repositioned:
   []
    [a] = "AAA"
   ```

```
[b] = "BBB"
[c] = "CCC"
```

3. It is no longer possible to use `vStruct2->b` because the `$parent` assignment has moved the Struct from `vStruct2` to `vStruct1`. A more elegant alternative is to use a dedicated reference by assigning `vStruct2->b` to its own variable. The end result is the same: `vStruct1`is assigned the member bat position 2.

Related concepts

$parent
$index
Adding, Copying, Moving, and Replacing Struct Members