



Rocket Uniface Library 10.4

Creating and Deleting Structs

Structs are created using the `$newstruct` ProcScript function, but they can also be created implicitly. They are removed when there is no longer a parameter or variable that refers to them.

To create Structs, you can:

Creating Structs

1. Use the ProcScript function `$newstruct` to explicitly create a new, empty Struct.

```
vStruct = $newstruct
```

2. Assign a value to a `struct` variable or parameter.

```
vStruct1->title = "Users Guide"
```

`vStruct` now points to a newly created Struct with member `title`.

For more information, see [Assigning Values to Structs](#).

3. Assign a value to an allowed Struct function (`$name`, `$index`, `$parent`, or `$scalar`) or to the `$tags` Struct.

```
vStruct2->$name = "My Book"
```

`vStruct2` created and assigned a name.

4. Use one of the conversion functions to transform a complex data structure, such as XML, or a Uniface entity to a Struct.

```
params
  xmlstream pXml: in
  struct pStruct: out
endparams
xmlToStruct pStruct, pXml
```

`pStruct` now points to a (top level) Struct.

5. Assign an existing Struct to the member of another existing Struct.

```
xmlToStruct pStruct1, pXml
vStruct2->memberX = pStruct1
```

`memberX` is a new Struct, which is a copy of `pStruct1`.

6. To create or update a Struct member, assign a member to an existing Struct.

If the named Struct exists, it is updated. If it does not, it is created.

```
variables
  struct vStruct1
  struct vStruct2
endvariables
...
xmlToStruct vStruct1, pXml ; Create vStruct1
...
vStruct1->complexMemberX->memberY = 1 ; Create/update a scalar member
vStruct1->complexMemberX->memberZ = vStruct2 ; Create/update a struct member
```

Deleting Structs and Members


To delete a Struct, you need to ensure that there are no variables or parameters that refer to it. There is no ProcScript statement to delete a Struct. For local variables, this automatically happens when the function or operation completes.

The best way to explicitly remove a Struct is to assign an empty value to the **struct** variable that refers to a Struct.

Example: Assign a new value to the Struct variable that refers to the Struct.

```
variables
  struct vStruct1, vStruct2
endvariables
vStruct1 = $newstruct
vStruct1->x = "member x"
vStruct1=""
```

vStruct2 now has no members called x.

 **Note:** When a Struct variable is declared, it is implicitly initialized to Null, so assigning such a Struct variable to an existing Struct variable removes the original Struct. For more information, see [Null Values](#).

Example: Creating and Building a Struct

```
variables
  struct vStruct, vStructCollection
  string vNames
endvariables
vNames = "John ;Mary
;
Jane"
while (vNames != "")
  vStruct = $newstruct ; - Create an empty Struct
  vStruct->name = $itemnr(1, vNames)
  vStructCollection->*{-1} = vStruct ; - append to collection
  delitem vNames, 1
```

```
endwhile  
vStructCollection->$name = "people"  
putmess vStructCollection->$dbgstring ; - display result in message frame
```

Related concepts

[xmlToStruct](#)

[componentToStruct](#)

[\\$newstruct](#)

[Working With Structs: Code Examples](#)