

# Uniface 9: hard to handle handles collections

## hard to handle handles collections

Author: ulrichmerkel@web.de (ulrich-merkel)

After all these years, there was a reply on my wish <http://unifaceinfo.com/support-handles-occhandle-in-lists/#comment-11058>. It was a great experience that Mike Tailor took some time to discuss alternatives with me; thanks a lot for this new feeling. In his eMails, he mentioned an option (which is possible from 9.7 onwards):

*With structs it is easy to create such a list and use it  
To add a handle to a list you can use*

*vStruct->handle{-1} = \$occhandle(ENT) ; Add a collection handle*

*And to use the handle there is no need to first get it out of the list*

*vStruct->handle{1}->op() ; Execute the operation "op" for the first instance of member handle*

As most of us like me will be 9.6 based for a longer time, here some hints in my reply:

*thanks for making it clear the your example runs in 9.7+ only.*

*For productive implementation, it's still a long way before 9.7 will be the standard platform.*

*Looks like I have to follow the 9.6.04 suggestion:*

*You can use an entity with a non-database field of data type handle to act as a reference collection,  
by filling multiple occurrences with handles and by exposing a set of collection operations.*

I started some dITo investigations to provide a useable interface to provide associative storing of handles for 9.6 as well as 9.7. A look at the examples and some first spike showed that the struct manipulations seems always have to hardcode the texts. First experiments showed that it takes some nasty tricks to have struct manipulations in a more flexible way (using parameters). I will keep you posted on my findings. Some history: use case: in the OGF, we have some highlighting, but need the "oldocc" to remove the previous highlighting. As we have a list for each entity in \$entityproperties, it is possible to add some "oldocc", \$curocc to this list. With \$curocc position this ends in a setocc to the old occ, remove highlites, setocc to the new position, add highlite there which costs a lot of performance. And if there is a sort in between, the position has changed so you still have 2 occurrences looking like the active one. My hope using \$occhandle was to avoid all that setocc which causes a lot of changes "under the hood". But it showed in 2010 that lists were not able to hold handle references.