
Subject: Are handles garbage collected?

Posted by [chase](#) on Fri, 14 Jul 1995 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

If I lose a handle ID is the dynamic memory allocated to that handle forever lost?

This would appear to be the case that it is lost since handle IDs are not builtin data types but are stored in long integers. Hence references to handles can not be kept track of (unless all long integer values are checked before they are overwritten - a very unlikely implementation). If I store a handle ID in a single location and then overwrite that location the ID is lost. As a result the memory can never be freed with `handle_free`.

It would seem that handle usage could be a source of memory leaks in programs.

I like the concept of handles to create heterogeneous arrays/lists. However, I am reluctant to use them wholesale in code manipulating large amounts of data. Memory leak problems could be very difficult to debug. Probably my most difficult and common bugs in C programs are memory management problems associated with pointers. It would appear that in IDL it is up to the user to manage handle memory - reintroducing my greatest difficulty from C into IDL.

It would be best if handles were builtin data types so that their memory would automatically be managed via garbage collection or reference counting. In fact, references would only need to be checked for top level handles.

I suppose the same memory leak problem is true with widgets that are created but never realized and their IDs are lost. Although this only happens when there is a bug preventing the execution of the code that realizes and manages widgets. To prevent memory problems without user intervention would require making widget IDs builtin data types, probably the same as handles.

Chris

--

=====

Bldg 24-E188

The Applied Physics Laboratory

The Johns Hopkins University

Laurel, MD 20723-6099

(301)953-6000 x8529

chris.chase@jhuapl.edu
