
Subject: Re: Treating an object as a structure
Posted by [JDS](#) on Fri, 06 Mar 2009 22:36:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

I use this technique a lot:

```
snapshot=create_struct(NAME=obj_class(self))
```

or simply

```
snapshot={MYCLASS}  
struct_assign,self,snapshot
```

later apply snapshot:

```
struct_assign,snapshot,self
```

This allows me to make snapshots of object data for the purpose of undoing changes, or examining how it was at a certain point in time. Keep a pointer to a list of these, and you have multiple undo. Add a text phrase when snapshotting ("Frobnoid change") and you can advertise "Undo Frobnoid change" and so on.

The lack of garbage collection starts getting painful at this point though (since older versions could refer to data newer versions have deleted).

Remember that snapshots of heap pointers (at whatever level in the heirarchy) are not immutable. If you want a deep copy, you can use the save/restore trick, first calling the new HEAP_NOSAVE routine on items you won't need a deep copy of. Or you can simply replicate certain heap data by hand (new=ptr_new(*old)). This reminds me that we're still lacking deep copy capability without hitting the disk.

JD
