Subject: Nasty compatibility issue in lists and hashes (and maybe more?)

Posted by penteado on Thu, 26 May 2011 02:51:47 GMT

After some time trying to figure out what was going on, I eventually
found that the changes in some classes between 8.0.1 and 8.1 break
compatibility of some savefiles, in both ways: not only 8.1 savefiles
cause trouble when restored in 8.0.1, but also 8.0.1 savefiles cause
trouble when restored in 8.1

One example:

IDL> print,!version

{ x86_64 linux unix linux 8.0.1 Oct 7 2010   64   64}

IDL> l=list(1,2,3)

IDL> h=hash(1,'a',2,'b',3,'c')

IDL> save,file='list_hash_8.0.1.sav',/verbose

% SAVE: Portable (XDR) SAVE/RESTORE file.

% SAVE: Saved variable: H.

% SAVE: Saved variable: L.

So far, so good. But then, from 8.1:

IDL> print,!version

{ x86_64 linux unix linux 8.1 Mar 9 2011   64   64}

IDL> restore,'list_hash_8.0.1.sav',/verbose

% RESTORE: Portable (XDR) SAVE/RESTORE file.

% RESTORE: Save file written by <unknown>@<unknown>, Wed May 25
23:41:49 2011.

% RESTORE: IDL version 8.0.1 (linux, x86_64).

% RESTORE: Tag names disagree. Using original definition:
IDL_CONTAINER_TOP(restored) ->

        IDL_OBJECT_TOP(existing).

% RESTORE: Unable to restore structure, tag type disagrees with
existing definition: LIST.

!NULL

  ^

% HASH is already defined with a conflicting definition.

% Wrong number of tags defined for structure: IDL_OBJECT.

% RESTORE: Structure not restored due to conflict with existing
definition: IDL_OBJECT.

% RESTORE: Restored variable: H.

% RESTORE: Restored variable: L.

IDL> help

% At $MAIN$

H   OBJREF   = <ObjHeapVar9>

L   OBJREF   = <ObjHeapVar2>

Compiled Procedures:

   $MAIN$

Compiled Functions:

Now I am wondering how to work around this: how can I make savefiles
in one version to be read in the other? I do not yet see a way, and

this is a serious problem. I have many applications based on savefiles, and lists and hashes are the most common objects in my recent savefiles.

For now, it seems I will have keep to 8.0.1 when using such savefiles, which would be very frustrating, negating all the important bug fixes and new features of 8.1 (I am thinking particularly of those in lists and hashes, which I had already been using through my derived classes).