
Subject: Re: Catalyst CatDestroyDefaults
Posted by [dosfun](#) on Wed, 24 Jun 2009 01:30:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

> Crane writes:
>> I am trying to use CatSetDefault to set some of my system settings,
>> for instance:
>
>> oLang = Obj_New('languageclass')
>> CatSetDefault, 'language', oLang
>
>> When my application is destroyed, the CatDestroyDefaults is called
>> automatically, I expected this procedure will clean up all system
>> settings, so I did not destroy the oLang object by myself.
>> Unfortunately, oLang is not cleaned up, it is still in memory. Should
>> I have to explicitly destroy the system settings in cleanup method ,
>> or where am I wrong?
>
>> I try to find what happens using debug mode, and know the
>> CatDestroyDefaults procedure finally go to the CatContainer::Cleanup
>> method, this method remove all child objects so they don't get
>> destroyed automatically, I think this is why oLang is not destroy.
>
>> I do not understand this, because we need CatDestroyDefaults to
>> automatically destroy all the system settings when we destroy the
>> application.
>
> It is probably appropriate that this comes on the heels of
> another discussion a week or so ago about the inability of
> UNDEFINE to undefine everything. And the reason for the problem
> is similar. It never occurred to me to store a heap variable
> in a Catalyst default value. ;-)
>
> But, now I know someone wants to do that, the solution is simple.
> If the internal pointer points to a heap variable, destroy that, too.
> And since I already spent an hour getting UNDEFINE to do just that
> a week ago, I decided to use that to get the job done.
>
> So, be sure you have the latest UNDEFINE, and you can download
> a new CatListValue object here:
>
> <http://www.dfanning.com/programs/undefine.pro>
> [.../programs/catalyst/source/core/catlistvalue__define.pro](http://www.dfanning.com/programs/catalyst/source/core/catlistvalue__define.pro)
>
> Tested only lightly, as I am late for work, as usual. :-)
>
> Cheers,

>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>

>

Very thanks, wonderful.
