How to implement the callback to application functions in a driver

Source:
http://www.tech−archive.net/Archive/WindowsCE/microsoft.public.windowsce.embedded/2005−07/msg00045.html

- *From*: Neverhoodboy <Neverhoodboy@xxxxxxxxxxxxxxxxxxxxxxxx>
- *Date*: Thu, 7 Jul 2005 01:16:04 −0700

Suppose I have the following data structure:

```
typedef struct {
void* data_ptr;
void* notify_param;
void (*notify_func)(void* param);
} ExampleStructure;
```

Here is what the application looks like:

```
void notification(void* param)
{
/* Does not have to be free(), user can specify the parameter and do
whatever he wants */
free(param);
}

void main()
{
void* data_ptr = malloc(1024);
ExampleStructure example = {data_ptr, data_ptr, notification};
DeviceIoControl(hDevice, &example);
while (1) {}
}
```

In the driver, I use GetOwnerProcess() and MapPtrToProcess() to map the example.data_ptr, and then do something with the data pointed by that pointer. When done, I call the example.notification (the address is also mapped) to notify the application that I'm done and let application do whatever it needs, usually a free(). However, what I've experienced is that if I do something as simple as operating the local variables in the notification function, it works, however, if I put a call to free() or access any global variable of the application, it generates data abort exception. Does anyone know how to solve this? Thanks a bunch!!!
.

- ***Follow−Ups***:
  - ***RE: How to implement the callback to application functions in a driver***
    - ◊ *From:* Neverhoodboy

- Prev by Date: ***Re: hard disk can not always be recognized on windows ce net 4.2***
- Next by Date: ***Re: index problem***
- Previous by thread: ***hard disk can not always be recognized on windows ce net 4.2***
- Next by thread: ***RE: How to implement the callback to application functions in a driver***
- Index(es):
  - ***Date***
  - ***Thread***