

# Re: Dealing with HANDLES

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.kernel/2006-12/msg00035.htm>

---

- *From:* "Pavel Lebedinsky [MSFT]" <pavel@xxxxxxxxxxxxxxxxxxxxxxxx>
  - *Date:* Fri, 1 Dec 2006 00:28:03 -0800
- 

"Arnaud Debaene" wrote:

Think more about it : Are you sure you want your "Process" class to be copiable? A process is a unique entity, with it's own identity (it's ProcessID), which can't be duplicated : the closest thing to copying a process is forking it, which is quite unnnatural on Windows and is NOT a mere copy.

It should be the same thing with your class : it shouldn't be copiable In that case, making it's copy constructor private and undefined should do the trick.

Now, if your class is more a "ProcessInformation" than a "Process", and you really want to be able to copy it, DuplicateHandle is the way to go (in both the copy constructor and the asignement operator).

Duplicated handles are not necessarily interchangeable. Each handle has its own flags that can be set with SetHandleInformation for example. The actual numeric value of the handle is also often used for things like tracing/debugging, or even passing it as a command line argument to child processes (for inheritable handles).

So I don't think using DuplicateHandle in a copy constructor or operator= is a good idea. I would just make this class non-copyable.

—  
This posting is provided "AS IS" with no warranties, and confers no rights.

.