Subject: Re: Shared Memory in Windows
Posted by Robbie on Thu, 06 Apr 2006 02:11:11 GMT
View Forum Message <> Reply to Message

Thanks for the heads up on the Windows page file. I ran some test code to observe it using pagefile.sys when the segment was larger than RAM. It's very comforting to see the theory match the practical results.

> The first rule of performance tuning is to only tune when
> you have to. Try it out -- you may find that the pagefile works fast
> enough for your needs.

Well, I'm actually investigating shared memory so I can make a decision on my data model. I'm mid-way through my project and I have deliberately left the data model quite open. However, I can't leave it open forever or else I will start to loose sleep.

I'm interested in using various C/C++ libraries such as ITK to perform image filtering and registration. A significant incompaibiltiy is that ITK uses data pipelines rather than committing pixeldata to memory. My own IDL code actually implements a poor mans version of data pipelines, with a mid-sized granularity (currently frame by frame because it works well with IDLgrWindow::QueryRequiredTiles). After writing a very butchered implementation of pipelines in IDL, I can see this project spinning out of control when I try to write C wrappers for the ITK data model. I also have to realize that ITK isn't the only imaging library which I might want to use.

Shared memory allows me to be a bit more generic and it makes design and testing much easier. The downside is that the space used by all opened images cannot exceed the size of RAM (At least with POSIX anyway). This is problematic if the UI keeps a track of intermediate images.

Robbie