

If you want to consume all the virtual address space, well, then go ahead and consume it, you don't need my help

 devblogs.microsoft.com/oldnewthing/20090629-00

June 29, 2009



Raymond Chen

Commenter [Matthew Chaboud](#) asks if there's an easy way to consume all the virtual address space below 4GB, short of, well, actually allocating it. "It seems like there should be a cleaner way to do this."

If you want to consume all the virtual address space, then call `VirtualAlloc` until you turn blue. Programs shouldn't care what address they get back from a memory allocation function; they should handle values below 2GB and above 2GB with equal facility.

It's not like there's a `ConsumeAllAvailableVirtualAddressSpaceAndExhaustTheHeap` function. (Is there a

`AllocateAllRemainingDiskSpaceAndFillExistingFilesWithZeroes` function?) What would be the point of such a function? Once you call it, you have run out of memory!

If Mr. Chaboud is talking about keeping programs away from bottom 4GB of virtual address space on a 64-bit machine, then a much easier way to do this is to set the AllocationPreference configuration setting to specify that memory should be allocated from high addresses first. (But I don't think that's the scenario that prompted the original question, because on 64-bit Windows, the default heap is above the 4GB boundary, so there would be no need to exhaust the heap in order to consume the memory at virtual addresses below 4GB.)

Correction: [Pavel Lebedinsky points out](#) that the default heap is below 4GB on 64-bit machines. It used to be above the 4GB boundary on earlier versions of 64-bit Windows, but I guess they changed it.

[Raymond Chen](#)

Follow

