Virtual allocation granularity and page size

Asked 15 years, 11 months ago   Modified 15 years, 11 months ago

Viewed 9k times

▲

12

▼

What are the typical values of the virtual allocation granularity and page size on Win64 platforms? That'd be SYSTEM_INFO's dwAllocationGranularity and dwPageSize.

On Win32 systems these would be 64k and 4k.

I need to know because I've designed a custom allocator based on VirtualAlloc for a Win32 application and wonder if my design choices are still valid on Win64. I have no access to a Win64 system.
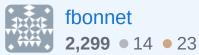
Thanks in advance!

`c` `windows` `winapi` `win64`

Share

Improve this question

Follow

edited Jan 13, 2009 at 13:09

asked Jan 13, 2009 at 13:03

fbonnet
2,299 ● 14 ● 23

Do these design choices depend on these specific constant values, or do they dynamically query dwAllocationGranularity and dwPageSize? In other words, could you share more context? – reuben Jan 14, 2009 at 5:47

1   The code queries the values. However I wonder about the impact on the application. For example, if the granularity is sufficiently small I can allocate dedicated address ranges to distinct modules. OTOH if it is too large (several MB) this may exhaust the address space so sharing them is necessary. – fbonnet Jan 14, 2009 at 14:59

1 Answer

Sorted by: Highest score (default) ⇕

It is still the same, 64k and 4k on Vista x64. Not so sure if that isn't going to change some day, the small page size is putting serious pressure on the TLB cache, degrading perf considerably for programs that allocate multi-gigabyte chunks. I've also seen several problem reports about not being able to allocate large pages anymore at some inscrutable point during program execution. But that's just crystal-ball staring for now.

11

Share   Improve this answer

Follow

answered Jan 13, 2009 at 13:59

Hans Passant
940k ● 148 ● 1.7k ● 2.6k

1   I've heard that page size was 8k on Itanium. Anyway, thanks for your answer! – fbonnet Jan 13, 2009 at 15:37

Yeah, 8K on IA64. Ever seen one? – Hans Passant Jan 13, 2009 at 18:03

1 Itanium - it is being retired by Intel. Just be glad you never had to work on one. I did when Windows XP was codenamed "Whistler". The box was a pre-production from Intel (and got replaced a few times with a "better" one) with an early version of Visual Studio 7.0. Slow, Noisy as hell, like having a loud vaccuum cleaner next to your desk all day. I ported a Win32 app to Win64 on it, 2,000,000 lines of C++. I was glad when that part of my job was complete and I didn't have to use the Itanium anymore. – Stephen Kellett Apr 10, 2010 at 13:32