Very Large Memory Support for Windows NT V5.0

Introduction

With Windows NT V5.0, Microsoft will be delivering support for process address space extensions. This new capability has been referred to in the press as "64-bit Windows NT." As indicated by the title of this white paper, the functional improvement is to add support for applications to effectively utilize Very Large Memory (VLM) physical memory configurations.

The purpose of this note is to communicate the conceptual nature of the extended address space support. It is not intended to provide complete technical details of the new support, and therefore is not a programming guide.

Windows NT V4.0

In Windows NT V4.0, a process address space can be up to 2GB of virtual address space. This address space contains the program code (EXE and DLLs), and program data, including local storage, thread-local storage, and stacks.

Note that although Windows NT V4.0 is a 32-bit operating system, it has some 64-bit capabilities built into it already:

- 64-bit integer arithmetic—Very large integers can be easily manipulated
- Support for files greater than 4GB
- Asynchronous I/O using 64-bit file offsets

Windows NT V5.0—Very Large Memory Address Space

In Windows NT V5.0, a new process virtual address space will be available for processes. This new VLM address space can only contain data (e.g., no code), and the set of APIs that operate on this data is restricted. The set of supported APIs is sufficient for developing programs that use large in-memory data store.

Windows NT 64-bit support provides VLM virtual address space in a fully upward compatible way, such that existing Win32 applications or drivers need no changes in order to run on Windows NT V5.0. Applications will only need modification for V5.0 in order to explicitly take advantage of the VLM capabilities.

Windows NT V5.0 is expected to enter into Beta test phase during the first half of CY1997, with a projected final product release by the end of CY1997. As the owner of Windows NT operating system technology, Microsoft will determine the exact dates and conditions under which V5.0 will be tested and delivered. Windows NT V5.0 is planned to have a number of other significant enhancements; information on those enhancements can be found on Microsoft's web page (http://www.microsoft.com).

VLM Operations Supported

The following operations are implemented in support of the VLM address space:

- **Allocate**—Make virtual address space available in the new data space.
- **Deallocate**—Deallocate some or all of the allocated virtual address space.
- **Change Page Protection**—Make pages read-only, read-write, etc.
- **Retrieve Memory Status**—Determine the status of allocated virtual address space.
- **Bulk Memory Access**—Fast copy, move, block fill, and block zero operations.
- **Standard Memory Access**—The C/C++ compiler will support a new pointer type that provides seamless C/C++ access to data in the VLM address space, as well as copying data between standard and VLM address spaces.
- **Debugging Support**
  - Read Memory, Write Memory—Provides access to the VLM address space from a debugger running in another process
- **Input/Output**
  - Read, Write—Perform disk I/O into or out of the VLM address space

The key point to remember is that this new capability is NOT an implementation of a Win64 API. That is, Windows NT VLM support is not an implementation of the current Win32 support with all addresses (and hence all pointers) extended to 64 bits. Rather, it enables a specific set of new API capabilities that NT applications can use to effectively utilize physical VLM configurations.

Restrictions and System Requirements for Utilizing Windows NT VLM

The Windows NT V5.0 VLM support has several specific characteristics and requirements:

- **Locked-down physical memory—**Memory addressed by 64-bit pointers is accessed using virtual addresses, but is backed by locked-down physical memory. The rationale for this restriction is very simple. The intent is to provide high speed access to very large data caches. The goal is to solve real problems for a small class of demanding, data intensive applications that require high speed access to their cached data sets.
- **Native 64-bit microprocessor requirement—**This new capability is supported only on processors that directly support native 64-bit addressing. Specifically, this includes the existing and future Digital Alpha microprocessor family. Systems currently being sold today with Windows NT V4.0 and sufficient physical memory capacity are expected to be software upgradeable. This requirement specifically excludes the current x86 Pentium and PentiumPro microprocessor family. Intel has

stated that the next generation *Merced* architecture will be a 64-bit design, and that IA-64 will first become available in systems in 1998, at the earliest. Microsoft has publicly stated that the Alpha architecture will first utilize the VLM capabilities, followed later by the IA-64 architecture.

- **Physical memory requirement—**A minimum of 2GB physical memory is required to enable VLM support. For development and testing purposes, there will be a way to enable it on systems with less than 2GB of physical memory.

Looking Beyond Windows NT V5.0

Eventually, complete Win64 support will be implemented in Windows NT. Of course, it is too early to accurately predict when this might appear, but an educated guess would put this in the 2001-2002 timeframe.

At that time, the capabilities described in this document will no longer be needed to support 64-bit applications. The above APIs will only be available in the Win32 support and would not be carried forward to the Win64 API. Nonetheless, applications built using the above APIs will continue to run unchanged until such time as the application is restructured to take advantage of the complete Win64 support.

Summary

The Windows NT V5.0 VLM support will provide an easy-to-use, extensible way for applications to manipulate very large data spaces. As software developers use this capability to enhance their most demanding, resource-hungry applications, they will achieve significant performance enhancements over current versions.

This will create a Windows NT-based operating environment capable of providing customers with greater performance by using VLM-enabled applications. These applications will enable customers to implement Windows NT-based solutions in more demanding, larger “Enterprise class” information technology solutions than the performance scalability provided by today’s Windows NT V4 systems.