

PTC® MKS Toolkit®

Porting Applications for Windows

PTC® MKS Toolkit®

PTC MKS Toolkit can make it easy for you to migrate or port software from UNIX/Linux systems to Windows systems. In addition to the complete development and interoperability environment discussed above, PTC MKS Toolkit development products include over 2700 UNIX APIs necessary to run UNIX/Linux applications and over 300 commands and utilities necessary to run scripts from systems. The combination of the PTC MKS Toolkit UNIX APIs and the command-line development environment on Windows reduces the complexity of a port from UNIX to Windows to the complexity that many developers are already used to: that of porting from one implementation of UNIX to another. The basic porting process includes:

- Gaining access to the source code on Windows.
- Examining make files, header files, and source code for porting issues.
- Compiling and linking.
- Fixing linkage issues.
- Debugging.
- Porting scripts.
- Integrating with Windows.
- Distributing the application.

Porting to 64-bit Windows

PTC MKS Toolkit for Enterprise Developers 64-bit Edition simplifies the migration of 32-bit and 64-bit UNIX applications to 64-bit Windows enabling you to deploy your UNIX/Linux applications on both Extended Architecture (for example, AMD64) and Itanium-based platforms. 64-bit Windows is an ideal platform for applications that require large amounts of memory and high-performance such as digital content creation, electronic design automation, data warehousing, mechanical design automation, and advanced scientific and research applications. But migrating a UNIX application to 64-bit Windows is not without issues. The PTC MKS Toolkit UNIX to Windows Porting Guide provides advice on dealing with these issues. In addition, the clean64 utility parses source code and cleans up pointer assignments that would result in a 64-bit pointer being assigned to a 32-bit data type. Finally, the Microsoft compiler can be quite useful in finding and handling other 32/64-bit assignment issues. In short, MKS Toolkit, working in conjunction with the Microsoft compiler, provides you with everything you need to successfully migrate UNIX applications to 65-bit Windows.

Porting X Applications

PTC MKS Toolkit for Enterprise Developers contains a standard X development environment based on X11R6 and Motif 2.1.30. It also contains the PTC X/Server software package which includes a full-feature X server and supporting applications. In addition, the PTC MKS Toolkit CD includes several sample X clients that you can compile with this development environment and use with any installed X server (such as PTC X/Server).

Porting Shared Libraries

Using the PTC MKS Toolkit migration products, you can easily port a C-language shared library for use with a PTC MKS Toolkit ported application. You can also build a shared library that is usable outside a PTC MKS Toolkit ported application, for example, linked with a native Windows application, dynamically loaded into a Visual Basic application, or implementing a COM object or ActiveX control. The PTC MKS Toolkit migration products include examples that demonstrate how to port a shared library to Windows using the product.

Porting Threaded Applications

The PTC MKS Toolkit migration products implement the POSIX threading API (PThreads) as defined in POSIX 1003.1 (1996) (formerly known as POSIX 1003.1c), along with a subset of the extensions added to the UNIX 98 specification (also known as Aspen Threads). POSIX threads are implemented on native Windows threads. This provides optimum interoperability with the Windows operating environment and significantly reduces the overhead for threads performance. However, Windows does not support all the features necessary to implement all the advanced optional POSIX threads features. Threads are kernel-level objects in Windows, and each thread is scheduled in relation to all other threads in the system. Windows 2000/XP/2003/Vista supports SMP hardware implementations, providing true concurrent threading. PTC MKS Toolkit implements all core POSIX thread functions, as well as the core thread-safe (re-entrant) functions, as specified in POSIX 1003.1 (1996).

Porting Daemons

PTC MKS Toolkit migration products make it easy to port a daemon process from a UNIX/Linux system to Windows with the PTC MKS Toolkit migration products. Windows refers to these processes as services. Windows provides a Service Control Manager that controls starting and stopping services. The PTC MKS Toolkit migration products come with a service framework for building services from your daemon programs, along with a copy of the UNIX inetd daemon ported using this framework. This framework lets you build a single service binary that works on virtually all Windows operating systems.

Evolving Ported Applications with COM

After porting your UNIX application to Windows using the PTC MKS Toolkit migration products, there are many ways to evolve your application to take more advantage of the Windows platform. For example, with PTC MKS Toolkit and Microsoft's reusable Component Object Model (COM) technology and you can create reusable COM objects from existing UNIX applications. To wrap UNIX/Linux code as a COM object, you must understand the concepts discussed in "Building Standalone DLLs" in the MKS Toolkit UNIX to Windows Porting Guide. You can, however, understand most of the COM concepts without reading about standalone DLLs.