

# OS/2 Warp's New Infrared Support

---

by *Steve Mastrianni*

In the past year, infrared devices have been popping up on notebooks, printers, pens, personal digital assistants (PDAs), and other devices. Since OS/2 Warp makes a great mobile platform, it was only natural that it should support this emerging technology.

The OS/2 Warp version of the IBM infrared support software consists of a 16-bit OS/2 physical device driver, [IRDD.SYS](#), and a 32-bit DLL, [LMDLL.DLL](#). The physical device driver is loaded with a `DEVICE=` statement in the `CONFIG.SYS` file, and the DLL must be installed in the `LIBPATH`. The infrared device driver is written for [OS/2 Warp Version 3](#) and will not work with earlier versions of OS/2.

The OS/2 Warp implementation supports serial asynchronous speeds of 9600, 19200, 38400, 57600, and 115200 bits per second (bps), referred to as "low speed," and a synchronous speed of 1152000 bps, referred to as "high speed." The low-speed device appears to the system as a 16550-like buffered Universal Asynchronous Receiver Transmitter (UART), while the high-speed device is a complex synchronous communications controller. The device driver is currently available for the IBM ThinkPad, and will be available soon in a generic version that will support most notebooks, dongles, and desktop infrared adapters.

The driver consists of two main portions: the IrLAP layer and the IrLMP layer. The IrLMP layer manages the higher-level connection and multiplexing logic, while the IrLAP layer manages the device-specific hardware. Together they implement the IrDA-compliant protocol described in the IrDA specification. The driver model is asynchronous in nature. That is, calls to the device driver return immediately to the caller. When the request is completed, the device driver calls back the application that made the request. The DLL exports 32-bit function calls to the link management API functions. The DLL does not export 16-bit calls.

Direct calls to the infrared device driver are not supported. To allow applications that utilize the infrared device to be easily written, the infrared DLL exports all of the necessary APIs to discover, connect, and communicate with other IrDA-compliant infrared devices. Thus, applications that need access to the infrared device need only issue a few Ring 3 APIs, and no knowledge of the underlying device driver is necessary.