É um desafio fornecer uma definição única e precisa para o termo driver. No sentido mais simples, um driver é um software que permite que o sistema operacional e um dispositivo se comuniquem um com o outro.

Por exemplo, suponhamos que um aplicativo de fotos precise ler algumas imagens de uma câmera fotográfica. O aplicativo terá que pedir uma solicitação ao sistema operacional, que, por sua vez, fará a solicitação ao driver.

O driver, que geralmente é desenvolvido pela mesma empresa que projetou e fabricou o dispositivo, sabe como se comunicar com o hardware para obter as fotos. Depois de o driver receber os dados da câmera, ele os devolve para o sistema operacional, que repassa para o aplicativo.

**Tradutor de informações**

Portanto, um driver é um software que traduz o que diz um hardware ou um dispositvo para que o computador possa entender. Sem um software de driver, o hardware conectado  (por exemplo, uma placa de vídeo ou impressora) não funcionará corretamente.

Mas um driver não é um processo e muito menos um programa gerido independentemente pelo sistema, mas sim um conjunto de tabelas contendo informações sobre cada periférico, bem como os fluxos de informação circulante entre um PC e um periférico.

Os drivers são entregues com o sistema operacional na maioria das vezes, podendo ser encontrados em aplicativos de atualizações (como o Windows Update), no painel de controle e por meio da verificação de atualizações. Se o sistema operacional não tiver o driver necessário, você pode verificar o disco que veio com o hardware ou dispositivo que deseja usar ou acessar o site do fabricante.

**Casos e casos**

Existem alguns produtos atualmente no mercado que dispensam a instalação de drivers, conhecidos como dispositivos “plug-and-play”. Como o próprio nome diz, basta apenas conectá-los ao PC através de uma porta USB para poder usá-los. Geralmente, HDs externos e máquinas fotográficas utilizam essa tecnologia.

Nem todo driver é escrito pela mesma empresa que projetou um dispositivo. Em muitos casos, um driver é programado de acordo com um padrão de hardware. Isso significa que ele pode ser escrito pela Microsoft (no caso do Windows), e, sendo assim, o fabricante do dispositivo não precisa fornecê-lo.

Outro detalhe importante é que nem todos os drivers se comunicam diretamente com dispositivos. Certas requisições (como a leitura de dados de um hardware) passam por diversos drivers que participam do processo, distribuídos em uma pilha de camadas.

Alguns drivers dessa pilha podem fazer modificações na solicitação. Estes não se comunicam diretamente com o dispositivo: eles simplesmente manipulam o pedido e repassam-no para os outros drivers.

Já alguns drivers funcionam como filtros, com o objetivo de observar e registrar informações sobre pedidos de aplicativos para dispositivos. Eles agem como verificadores para garantir que os outros drivers que estão na pilha estão lidando com o pedido corretamente.