

Cuadro resumen de las características de los diferentes procesos de traducción de programas fuente

	Compilación	Interpretación	Máquina virtual
Fichero Ejecutable	-Genera un ejecutable	-No genera un ejecutable	-Genera código intermedio, portable entre plataformas, dirigido a una CPU imaginaria (máquina virtual) que posteriormente hay que traducir a la máquina real
Veces proceso traducción	-El proceso de traducción se realiza una sola vez	-El proceso de traducción se realiza para cada instrucción en cada ejecución	-Se realiza una sola traducción a código intermedio, y una interpretación muy rápida del código intermedio en cada ejecución.
Rapidez ejecución	-La ejecución es muy rápida debido a que el programa ya ha sido traducido a código máquina	-La ejecución es más lenta, ya que para cada línea del programa es necesario realizar la traducción en el momento de la ejecución.	-La ejecución no es tan rápida como en la compilación tradicional ni tan lenta como en la interpretación.
Relación ejecutable con plataforma (portabilidad)	El compilador está programado para obtener ejecutables para una arquitectura determinada. Los ejecutables no son portables entre arquitecturas diferentes.	-No hay ejecutable, así que si existe un intérprete para varias plataformas concretas, el programa se podrá ejecutar en ellas. Típicamente, los programas interpretados son mucho más portables que los compilados, ya que suelen existir intérpretes del mismo lenguaje en distintas plataformas.	-El ejecutable va dirigido a una CPU imaginaria (máquina virtual). Se puede luego portar a una plataforma para la cual exista esa máquina virtual (el intérprete de código intermedio).
Potencia	-Los lenguajes compilados suelen proporcionar al programador mecanismos más potentes y flexibles, a costa de un mayor vínculo con la plataforma.	-Los lenguajes interpretados no suelen ser muy dependientes de la plataforma de destino, pero en contrapartida suelen ser menos flexibles y potentes que los compilados.	-La plataforma de destino es virtual. Así pues, los programas son dependientes de esta plataforma virtual, que luego son ejecutados sobre plataformas reales por la máquina virtual.
Confidencialidad ficheros fuente	-Una vez compilado el programa, el código fuente no es necesario para ejecutarlo, así que puede permanecer en secreto si se desea.	-El código fuente es necesario en cada ejecución, así que no puede permanecer en secreto	-El código fuente no es necesario para la ejecución, sólo el código intermedio.
Detección de errores	-Los errores sintácticos se detectan durante la compilación. Si el fichero fuente contiene errores sintácticos, el compilador no producirá un ejecutable.	-Los errores sintácticos se detectan durante la ejecución, ya que traducción y ejecución se van haciendo simultáneamente. Algún error sintáctico podría quedar enmascarado, si para una ejecución concreta no es necesario traducir el código que lo contiene (algunos intérpretes son capaces de evitar esta situación)	-Los errores sintácticos se detectan durante la compilación del código fuente a código intermedio.
Errores de ejecución	-Un programa compilado puede, por error, afectar seriamente a la estabilidad del sistema, comprometiendo la ejecución de los otros procesos, por ejemplo, acaparando la CPU, la memoria o algún otro recurso, siendo a veces complicado para el sistema operativo interrumpir su ejecución.	-Un programa interpretado con un comportamiento erróneo normalmente puede ser interrumpido sin dificultad, ya que su ejecución está bajo el control del intérprete y no sólo del sistema operativo.	-Un programa con un comportamiento erróneo es ejecutado sobre la máquina virtual, que tiene un control absoluto sobre él, con lo que no se suele comprometer la estabilidad de la plataforma real.