

Application_Binary_Interface

An **Application Binary Interface** (ABI) describes the low-level interface between an application program and the operating system, between an application and its libraries, or between component parts of the application. An ABI differs from an application programming interface (API) in that an API defines the interface between source code and libraries, so that the same source code will compile on any system supporting that API, whereas an ABI allows compiled object code to function without changes on any system using a compatible ABI.

The ABI (Application Binary Interface) is a standard developed by ARM and its partners

The EABI (Embedded Application Binary Interface) points to the same thing; it just highlights the fact that ABI is for **embedded** world. At the writing of this article, there are two compilers that can be used to compile against EABI in Symbian OS 9, i.e. RVCT (RealView Compilation Tools) and GCCE.

The format of the output of the EABI compiler is ELF (Executable and Linking Format). Symbian translates standard ELF format into Symbian specific format, called E32Image.