

Glossary

The following terms are used throughout this document to describe varying aspects of input localization:

Component	An executable image. Components defined in this specification support one of the defined module types.
CIS	Core Interface Specification.
DXE	Framework Driver Execution Environment phase.
EDK	EFI Developer Kit. (required to build the EFI-SCT)
EDK_SOURCE	An environmental variable that specifies the base directory where the EFI source files are located. Typically this directory will be C:\TianoCore\Edk.
EFI	Generic term that refers to one of the versions of the EFI specification: EFI 1.02, EFI 1.10, UEFI 2.0 or UEFI 2.1.
EFI 1.10 Specification	Intel Corporation published the Extensible Firmware Interface Specification. Intel has donated the EFI specification to the Unified EFI Forum, and the UEFI now owns future updates of the EFI specification. See UEFI Specification Version 2.x.
EMS	UEFI Management Side. UEFI SCT management side that acts as the management side in SCT passive mode. This OS-based manager side provides uniform GUI test case management, test case execution, test report generation capabilities.
Foundation	The set of code and interfaces that glue implementations of EFI together.
Framework	Intel® Platform Innovation Framework for EFI consists of the Foundation, plus other modular components that characterize the portability surface for modular components designed to work on any implementation of Intel's EFI Implementation architecture.
GUID	Globally Unique Identifier. A 128-bit value used to name entities uniquely. An individual without the help of a centralized authority can generate a unique GUID. This allows the generation of names that will never conflict, even among multiple, unrelated parties.
IHV	Independent Hardware Vendor.

IHV SCT

Independent Hardware Vendor software-certification test.

Module

A module is either an executable image or a library instance. For a list of module types supported by this package, see module type.

Module Type

All libraries and components belong to one of the following module types: BASE, SEC, PEI_CORE, PEIM, DXE_CORE, DXE_DRIVER, DXE_RUNTIME_DRIVER, DXE_SMM_DRIVER, DXE_SAL_DRIVER, UEFI_DRIVER, or UEFI_APPLICATION. These definitions provide a framework that is consistent with a similar set of requirements. A module that is of module type BASE, depends only on headers and libraries provided in the MDE, while a module that is of module type DXE_DRIVER depends on common DXE components. For a definition of the various module types, see module type.

Protocol

An API named by a GUID as defined by the EFI specification.

PEI

Pre-EFI Initialization Phase.

PPI

A PEIM-to-PEIM Interface that is named by a GUID as defined by the PEI CIS.

Runtime Services

Interfaces that provide access to underlying platform-specific hardware that might be useful during OS runtime, such as time and date services. These services become active during the boot process but also persist after the OS loader terminates boot services.

SAL

System Abstraction Layer. A firmware interface specification used on Intel® Itanium® Processor based systems.

SCT

Self-certification test.

SCT Agent

The overriding driver for self-certification tests. This is the driving software that allows tests to be run, then records test results and locks the results from modification.

SEC

Security Phase is the code in the Framework that contains the processor reset vector and launches PEI. This phase is separate from PEI because some security schemes require ownership of the reset vector.

Shell

EFI Shell is a simple, interactive “command interpreter” that allows EFI device drivers to be loaded, EFI applications to be launched, and operating systems to be selected for boot.

UEFI

Unified Extensible Firmware Interface

UEFI Application

An application that follows the UEFI specification. The only difference between a UEFI application and a UEFI driver is that an application is unloaded from memory when it exits regardless of return status, while a driver that returns a successful return status is not unloaded when its entry point exits.

UEFI Driver

A driver that follows the UEFI specification. Modular firmware code that supports chipset or platform features and is reusable in multiple system contexts. Code relating to a device or function that is run in the pre-OS phase of execution. (Formerly known as an option ROM.)

UEFI Specification Version 2.0

First version of the EFI specification released by The Unified EFI Forum. This specification builds on the EFI 1.10 specification and transfers ownership of the EFI specification from Intel to a non-profit, industry trade organization.

UEFI Specification Version 2.1

Current version of the EFI specification released by the Unified EFI Forum.

Unified EFI Forum

A non-profit collaborative trade organization formed to promote and manage the UEFI standard. For more information, see www.uefi.org.