SUPPORTING UEFI CAPSULE FIRMWARE UPDATES ON LEGACY BIOS

HP INC
Supporting UEFI Capsule Firmware Updates on Legacy BIOS

**Abstract:** A technique is disclosed that utilizes WMI and SMM to overcome the dependency of capsule firmware updates on a UEFI-specific ESRT table and runtime service so that capsule firmware updates can be supported on legacy BIOS as well as UEFI.
This disclosure relates to the field of operating systems.

A technique is disclosed that utilizes WMI and SMM to overcome the dependency of capsule firmware updates on a UEFI-specific ESRT ("EFI System Resource Table") table and runtime service. This technique allows capsule firmware updates to be supported on both UEFI and legacy BIOS.

A capsule firmware update is designed to be UEFI-only. It depends on the ESRT table for an interface and runtime service to transfer the control and firmware image from the OS to the pre-boot UEFI environment. However, there are still some OSs, Win7 or Linux for example, which boot from the legacy BIOS. So the definition in UEFI, and Microsoft's implementation on Windows, do not support capsule updates for systems which boot from legacy BIOS.

It is desirable to overcome the dependency of capsule firmware updates on UEFI so that the same firmware update package will work on both UEFI and legacy BIOS, while still maintaining the benefits of updating in a UEFI environment if present.

According to the present disclosure, and as understood with reference to the Figure, the disclosed capsule firmware update technique is shown within a client 10 running Windows 20 and a BIOS 30. The left hand side flow illustrates the existing Windows capsule update implementation 40. The right hand side flow illustrates the new capsule update implementation 50 of the present disclosure.

The new capsule update implementation 50 reads all information reported from the ESRT table via a private WMI command. It then builds the FW capsule images. The capsule images are saved to a FAT formatted disk or, if sufficient FAT formatted disk space is not available, to SMM memory in order to transfer the firmware image from the OS environment to the pre-boot UEFI environment. The process then, instead of calling a runtime service, calls private WMI to reset the system and initiate a firmware update after reset. In the case where the capsule images are stored to SMM memory, a capsule warm reset is issued so that the memory content is retained after the reset. Then the capsule images are retrieved and the firmware update is processed from this point in the same or similar manner as in the existing Windows capsule update implementation.

The disclosed technique advantageously provides a unified and complete firmware update solution for both UEFI and legacy BIOS systems. This simplifies development, reduces support costs, and provides a consistent user experience.

*Disclosed by Wei Ze Liu, Tom Hung, Nung-Kai Chen, and Yi-Fan Hsia, HP Inc.*