WHEA Generic Error Source

Implementation Notes

May 5, 2010

Abstract

This paper provides information about the Windows® Hardware Error Architecture (WHEA) for the Windows family of operating systems. It provides guidelines for server platform vendors to properly implement a generic error source.

This information applies to Windows Server® 2008 R2.

References and resources discussed here are listed at the end of this paper.

The current version of this paper is maintained on the Web at:
 <http://www.microsoft.com/whdc/system/pnppwr/whea/WHEA_GenericErrorSource.mspx>

***Disclaimer****: This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.*

*This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.*

© 2010 Microsoft Corporation. All rights reserved.

Document History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Change |  |  |  |
| May 5, 2010 | First publication |

Contents

[Introduction 3](#_Toc257367076)

[Generic Error Source 3](#_Toc257367077)

[Error Status Block 4](#_Toc257367078)

[Notification Methods 7](#_Toc257367079)

[Resources 7](#_Toc257367080)

# Introduction

The Windows® Hardware Error Architecture (WHEA) is the foundation for hardware error handling on Windows operating systems. WHEA provides interfaces for the platform and Windows to collaborate on hardware error identification, reporting and, if possible, recovery. Reporting hardware errors through WHEA enables the platform to use the features of the Windows operating system to provide error logging functionality and to integrate with management software. In addition, the collaboration between the platform and Windows for error recovery provides continuous system availability to the end user.

The notion of a hardware *error source* is a fundamental concept of WHEA. A hardware error source is any hardware unit that alerts the operating system to the presence of a hardware error condition. Examples of hardware error sources include processor machine check exceptions, chipset error signals, I/O bus errors, and I/O device errors. A single hardware error source might handle error reporting for more than one type of hardware error condition. For example, a processor’s machine check exception typically reports processor errors, cache and memory errors, and system bus errors.

WHEA includes low-level hardware error handlers (LLHEHs) for standard hardware error sources that exist on most platforms, such as those that were previously mentioned. WHEA is also extensible so that it can support other types of hardware errors or error reporting mechanisms through the use of a *generic error source*. A generic error source provides a mechanism for the platform to report these hardware errors to Windows. The generic error source support in Windows Server 2008 R2 complies with both the WHEA Platform Design Guide and revision 4.0 of the Advanced Configuration and Power Interface (ACPI) specification. This paper provides clarification to the information in the WHEA Platform Design Guide and the ACPI specification for implementing a generic error source.

For more information about WHEA, see “WHEA – Windows Hardware Error Architecture” on the WHDC website. See also the WHEA Platform Design Guide, which is available to Microsoft partners under NDA, and section 17 of the ACPI specification, revision 4.0.

# Generic Error Source

The platform enumerates its error sources, including the generic error source, by using the ACPI hardware error source table (HEST). The HEST is described in Table 4.1 of the WHEA Platform Design Guide and in Table 17.3 of the ACPI specification, revision 4.0. The platform uses a generic error source to implement one or more of the following types of error sources:

* An error source that notifies the operating system of the presence of an error condition by using a nonstandard notification mechanism. For example, such an error source might use a system control interrupt (SCI).
* An error source in which the information that is associated with an error condition is encoded in a nonstandard format.
* An error source for which the platform implements firmware-first handling of an error condition.

The implementation of a generic error source relies on a block of memory that is shared between the platform and Windows. This block of memory is called the error status block (ESB). The platform must allocate the memory for the ESB during system startup, before the operating system is started. This memory should be marked as *firmware reserved* in the memory map, which prevents the Windows memory manager from using this memory for any other purpose. WHEA obtains the information it needs to access the ESB from the Generic Hardware Error Source Structure that is in the HEST. This structure is described in Table 4-12 of the WHEA Platform Design Guide and in Table 17-11 of the ACPI specification, revision 4.0. This structure specifies where WHEA can obtain the 64‑bit physical memory location of the ESB and the length of the ESB.

**Note:** There are two levels of indirection to obtain the physical address of the ESB from the Generic Hardware Error Source Structure because the address information in the Error Status Address field of the structure points to a 64-bit register and the contents of that register points to the physical location of the ESB.

# Error Status Block

The format of the ESB is described in Table 4-13 of the WHEA Platform Design Guide and in Table 17-12 of the ACPI specification, revision 4.0. This paper discusses the semantics of the format of the ESB rather than the details about each field.

Figure 1 shows the layout of the ESB in memory. Note that the ESB is reused every time that the generic error source reports an error to WHEA.

Figure 1. Generic error source ESB

Figure 2 shows the layout of the ESB if no raw data is associated with the generic error source.

Figure 2. Generic error source ESB that has no raw data

The following provides information about each ESB field that will help you to correctly implement a generic error source.

Block Status

The Block Status field indicates what kind of error is reported in the ESB. The ESB always reports a single correctable or uncorrectable error condition. However, bits in the block status also indicate multiple correctable or uncorrectable errors. These bits are only for informational use. For every generic error notification that the platform sends to Windows, the error information that is reported in the ESB results in a single error record, regardless of the state of the multiple error bits. The number of generic error data entries in the ESB is also encoded in the Block Status field. When an error is reported in the ESB, there must be at least one generic error data entry in the ESB. The format of the Block Status field is described in Table 4-13 of the WHEA Platform Design Guide and in Table 17-12 of the ACPI specification, revision 4.0.

**Note**: An error exists in earlier versions of the WHEA Platform Design Guide and in section 17.3.2.6.1 of the ACPI specification, revision 4.0. Both documents state that the ESB can contain zero generic error data entries. This information is incorrect. The ESB must contain at least one generic error data entry.

Raw Data Offset

The Raw Data field is an optional field of the ESB that contains data that is opaque to Windows during error condition processing. This data is included in the WHEA error packet in its original raw form. The Raw Data Offset field contains the offset from the start of the ESB to the start of the raw data. If raw data is present in the ESB, it must follow all generic error data entries. In this situation, the Raw Data Offset field must be at least Data Length + 20. Note, however, that the raw data is not required to immediately follow the generic error data entries. This enables the platform to select the alignment of the raw data that is most convenient for the platform.

If no raw data is present in the ESB, the Raw Data Offset and Raw Data Length fields are both set to zero.

Raw Data Length

The length of the raw data must not exceed the value of the Max Raw Data Lengthfield of the Generic Hardware Error Source Structure. In addition, the value of the Raw Data Offset field plus the value of the Raw Data Length field must not exceed the value of the Error Status Block Length field of the Generic Hardware Error Source Structure. The Generic Hardware Error Source Structure is described in Table 4-12 of the WHEA Platform Design Guide and in Table 17-11 of the ACPI specification, revision 4.0.

If no raw data is present in the ESB, the Raw Data Offset and Raw Data Length fields are both set to zero.

Data Length

The Data Length field specifies the total size, in bytes, of all generic error data entries in the ESB. As previously stated, the ESB must contain at least one generic error data entry when an error condition is reported to Windows.

Error Severity

The Error Severity field of the ESB specifies the severity of the entire error event. Each generic error data entry in the ESB also includes an Error Severity field, which specifies the severity of the specific error data in the generic error data entry.

Generic Error Data Entries

As previously stated, the ESB must contain at least one generic error data entry when an error condition is reported to Windows. The format of the generic error data entries is described in Table 4-14 of the WHEA Platform Design Guide and in Table 17-13 of the ACPI specification, revision 4.0. All generic error data entries are reported in a single WHEA error record. The ability to include multiple generic error data entries for a single error condition enables the platform to include information from related components that are relevant to the error in the ESB. For example, if a PCI device indicates an error condition, the platform could use a separate generic error data entry to include information about the state of the PCI bridge for the particular PCI bus where the device is connected.

Raw Data

The Raw Data field is an optional field of the ESB that contains data that is opaque to Windows during error condition processing. This data is included in the WHEA error packet in its original raw form. The amount of raw data must not exceed the value of the Max Raw Data field of the Generic Hardware Error Source Structure. The Generic Hardware Error Source Structure is described in Table 4-12 of the WHEA Platform Design Guide and in Table 17-11 of the ACPI specification, revision 4.0.

If no raw data is present in the ESB, the Raw Data Offset and Raw Data Length fields are both set to zero.

# Notification Methods

The Notification Structure field of the Generic Hardware Error Source Structure specifies the notification mechanism that the platform uses for the generic error source to notify Windows about an error condition. This field contains a Hardware Error Notification Structure, which is described in Table 4-15 of the WHEA Platform Design Guide and in Table 17-14 of the ACPI specification, revision 4.0. When Windows receives notification of an error condition from the generic error source, it reads the Block Status field of the ESB to determine the type of error that occurred.

# Resources

ACPI Specification

<http://www.acpi.info>

WHEA – Windows Hardware Error Architecture

<http://www.microsoft.com/whdc/system/pnppwr/WHEA/default.mspx>

WHEA Platform Design Guide

Available to Microsoft partners under NDA.