

How can I determine the underlying cause of a EXCEPTION_IN_PAGE_ERROR exception?

devblogs.microsoft.com/oldnewthing/20120608-00

June 8, 2012



Raymond Chen

A customer was using memory-mapped files and installed an exception handler to log in-page errors in the memory-mapped file region. They wanted to know how they could obtain the real disk error that resulted in the memory manager not being able to page-in the requested data.

Finding the answer isn't that hard. A quick search for `EXCEPTION_IN_PAGE_ERROR` reveals that the information is provided in the `ExceptionInformation` member of the `EXCEPTION_RECORD` structure.

`EXCEPTION_IN_PAGE_ERROR`

The first element of the array contains a read-write flag that indicates the type of operation that caused the access violation. If this value is zero, the thread attempted to read the inaccessible data. If this value is 1, the thread attempted to write to an inaccessible address. If this value is 8, the thread causes a user-mode data execution prevention (DEP) violation.

The second array element specifies the virtual address of the inaccessible data.

The third array element specifies the underlying NTSTATUS code that resulted in the exception.

In other words,

```
if (GetExceptionCode() == EXCEPTION_IN_PAGE_ERROR)
{
    DiskError = GetExceptionInformation()->
                ExceptionRecord->
                ExceptionInformation[2];
}
```

Raymond Chen

Follow

