

Why does HRESULT begin with H when it's not a handle to anything?

 devblogs.microsoft.com/oldnewthing/20180117-00

January 17, 2018



Raymond Chen

I got a piece of email from a colleague who wondered why `HRESULT` begins with an H when it's not a handle to anything.

“The question came up at lunch today, and your name soon followed. We all just assume you have some kind of time traveling phone booth or Tardis, and like Bill and Ted of old, move through geek history to find answers to those long lost questions.”

As I understand it, in the old days it really was a handle to an object that contained rich error information. For example, if the error was a cascade error, it had a link to the previous error. From the result handle, you could extract the full history of the error, from its origination, through all the functions that propagated or transformed it, until it finally reached you.

For example, if a function receives an object from the client, and that object is expected to support a particular interface, but when the function asks the client for that interface, it gets the error “interface not supported”, the function would transform the error to “invalid parameter” because the client passed an invalid object. When the caller receives the error object, it can unpack the error information to determine, “Okay, I got an *invalid parameter* error because the *DoSomething* method received an *interface not supported* error on its second parameter. Hey, that's me!”

It's great having all of this information available, but it comes at a cost. Everybody who generates or propagates or transforms errors needs to create or update the error object to include their piece of the puzzle. And every function call that consumes an error and decides not to propagate or transform it must remember to free the error object.

In practice, programs don't care about the nature of the error to this level of detail. They just want to know that an error occurred, and get a general idea of what kind of error it was. There's not much point creating all this detailed error information, only to have the app throw it away anyway.

The COM team decided that the cost/benefit simply wasn't worth it, so the **HRESULT** turned into a simple number. But the name stuck.

Raymond Chen

Follow

