

What kind of caller diagnostic information can I get from exceptions thrown by C++/WinRT and wil?, C++20 edition



Raymond Chen

A short time ago, I summarized the kind of caller diagnostic information you can get from exceptions thrown by C++/WinRT and wil. My colleague David Machaj reminded me that the story had improved in the time since I originally wrote the article, so it's time for an update.

Starting in version 2.0.220929.3 of C++/WinRT, if `std::source_location` is supported, then C++/WinRT uses it to pass file and line number information to wil when a C++/WinRT exception is thrown. In practice, this happens when you compile in C++20 mode or later.

The revised table now looks like this:

	C++/WinRT			wil	
	no <code>wil/cppwinrt.h</code>	with <code>wil/cppwinrt.h</code>		no C++/CX	with C++/CX
		<code>throw_hresult_error</code>	<code>check_hresult</code>	<code>THROW_IF_FAILED</code>	<code>THROW_IF_FAILED</code>
Thrown type	<code>hresult_error</code>	<code>hresult_error</code>	<code>hresult_error</code>	<code>Result-Exception</code>	<code>Exception^</code>
Stack trace in thrown object	Yes	Yes	Yes	No	Yes
Stack trace in thread data	Yes	Yes	Yes	Requires <code>result_originate.h</code>	Yes
File/line number in thrown object	No	No	No	Yes	No

Exception in wil error log	No	No	Yes	Yes	Yes
File/line number in wil error log	No	Requires C++20	Requires C++20	Yes	Yes

Raymond Chen

Follow

