

# How can I include/exclude specific memory blocks in user-mode crash dumps?

 [devblogs.microsoft.com/oldnewthing/20181011-00](https://devblogs.microsoft.com/oldnewthing/20181011-00)

October 11, 2018



Raymond Chen

You can tweak the information included in crash dumps created by Windows Error Reporting.

To request that Windows Error Reporting include a dump of another process if the calling process crashes, use [the `WerRegisterAdditionalProcess` function](#). This is useful if your program is split up into multiple processes.

You can add additional key/value pairs of data with [the `WerRegisterCustomMetadata` function](#). This information can be used to help categorize and filter crash dumps automatically. For example, you might include metadata that says whether the app is running in trial mode.

You can add a block of memory to user-mode crash dumps (heap dumps or larger), with [the `WerRegisterMemoryBlock` function](#). Note that this memory will be captured in the dump, but it's not the same as metadata because you can't filter on it.

Conversely, you can exclude a block of memory from user-mode crash dumps with [the `WerRegisterExcludedMemoryBlock` function](#). This is handy if you have large memory blocks containing information that isn't all that interesting from a debugging standpoint, like video texture buffers or audio output buffers.

[Raymond Chen](#)

**Follow**

