Why does the access violation error message put the operation in quotation marks? Is is some sort of euphemism?

April 3, 2014

Raymond Chen

When an application crashes with an access violation, the error message says something like

> The instruction at "XX" referenced memory at "YY". The memory could not be "read".

Why is the operation in quotation marks? Is this some sort of euphemism? The odd phrasing is a consequence of globalization. The operation name is a verb in the infinitive ("read", "write"), but depending on how the containing message is localized, it may need to take a different form. Since the kernel doesn't understand grammar, it just puts the words in quotation marks to avoid having to learn every language on the planet. Imagine if it tried:

> The memory could not be readed.

The kernel tried to form the passive, which is normally done in English by adding "−ed" to the end of the verb. Too bad "read" and "write" are irregular verbs! The more conventional solution for this type of problem is to create a separate error message for each variant so that the text can be translated independently. rather than building sentences at runtime,

The access violation error message is in a pickle, though, because the underlying status code is `STATUS_ACCESS_VIOLATION`, and that message contains three insertions, one for the instruction address, one for the address being accessed, and one for the operation. If there were three different status codes, like `STATUS_ACCESS_VIOLATION_READ`, `STATUS_ACCESS_VIOLATION_WRITE`, and `STATUS_ACCESS_VIOLATION_EXECUTE`, then a separate string could be created for each. But that's not how the status codes folks decided to do things, and the translation team was stuck having to use the ugly quotation marks.

Raymond Chen

Follow