

Why can't I get FormatMessage to load my resource string?

devblogs.microsoft.com/oldnewthing/20080229-00

February 29, 2008



Raymond Chen

A common mistake I see is people confusing message tables with string resources. This basically shows up in one place: `FormatMessage`.

The folks who were designing Win32 decided that plain string resources weren't fancy enough, so they invented message tables, which is just another way of storing strings in resources. Why string resources weren't good enough I don't know.

The `FormatMessage` function accepts a message number, which is looked up in a message table, not in a string resource. In other words, if you have something like this:

```
#define IDS_MYMESSAGE 100
STRINGTABLE BEGIN
    IDS_MYMESSAGE "Hello, %1"
END
TCHAR szBuffer[256];
FormatMessage(FORMAT_MESSAGE_FROM_HMODULE, g_hInstance,
             IDS_MYMESSAGE, 0, szBuffer, 256, Arguments);
```

the call to `FormatMessage` will fail because you passed in a string resource identifier, not a message identifier. Since nobody actually uses message tables, you almost certainly want to pass the `FORMAT_MESSAGE_FROM_STRING` flag.

```
TCHAR szMessage[256];
LoadString(g_hInstance, IDS_MYMESSAGE, szMessage, 256);
TCHAR szBuffer[256];
FormatMessage(FORMAT_MESSAGE_FROM_STRING, szMessage,
             IDS_MYMESSAGE, 0, szBuffer, 256, Arguments);
```

The only place you will normally want to format a message from a message table is when you want to access the system message table to get the text associated with an error code. (But don't forget the `FORMAT_MESSAGE_IGNORE_INSERTS` flag!)

Raymond Chen

Follow

