

# Why can't I use PSGUID\_STORAGE like a GUID?

 [devblogs.microsoft.com/oldnewthing/20110829-00](http://devblogs.microsoft.com/oldnewthing/20110829-00)

August 29, 2011



Raymond Chen

The `stgprop.h` header file defines a GUID called `PSGUID_STORAGE`, but a customer was having trouble using it.

```
GUID guid;
...
// This generates a strange compiler error
if (IsEqualGUID(guid, PSGUID_STORAGE)) { ... }
```

The strange compiler error the customer referred to is the following:

```
test.cpp(136) : error C2143: syntax error : missing ')' before '{'
test.cpp(136) : error C2059: syntax error : ')'
test.cpp(136) : error C2143: syntax error : missing ';' before '{'
test.cpp(136) : error C2059: syntax error : '{'
test.cpp(136) : error C2059: syntax error : ')'
test.cpp(137) : error C2059: syntax error : '}'
test.cpp(137) : error C2143: syntax error : missing ';' before '}'
test.cpp(137) : error C2059: syntax error : '}'
```

“I don’t see what the compiler is complaining about. The parentheses appear to be properly matched before the left brace.”

Remember, what you see is not necessarily what the compiler sees. Let’s take another look at this mysterious GUID:

```
#define PSGUID_STORAGE { 0xb725f130,          \
                        0x47ef, 0x101a,      \
                        { 0xa5, 0xf1, 0x02, 0x60, 0x8c, 0x9e, 0xeb, 0xac } }
```

Well there’s your problem. After the preprocessor does its substitution, the line becomes

```
if (IsEqualGUID(guid, { 0xb725f130,
                        0x47ef, 0x101a,
                        { 0xa5, 0xf1, 0x02, 0x60, 0x8c, 0x9e, 0xeb, 0xac } }))) { ... }
```

and that’s not legal C/C++. (Though with a little tweaking, [you can get GCC to accept it.](#)) The `PSGUID_STORAGE` symbols is intended to be used as an initializer:

```
const GUID StorageGuid = PSGUID_STORAGE;
```

“How did you know that?”

I didn't, but I went to the effort of looking at the definition in the header file and figuring it out from inspection.

Why is it defined this way instead of

```
DEFINE_GUID(PSGUID_STORAGE, 0xb725f130, 0x47ef,  
            0x101a, 0xa5, 0xf1, 0x02, 0x60, 0x8c, 0x9e, 0xeb, 0xac);
```

?

Because this GUID is used as the FMTID of a PROPERTYKEY. The PROPERTYKEY structure looks like this:

```
typedef struct {  
    GUID fmtid;  
    DWORD pid;  
} PROPERTYKEY;
```

The intended usage is evidently

```
const PROPERTYKEY  
PKEY_STORAGE_DIRECTORY = { PSGUID_STORAGE, PID_STG_DIRECTORY };
```

Since the C language does not permit global variables to be initialized from other global variables (or at least it didn't at the time `PROPERTYKEY` s were defined; who knows what crazy features will show up in C1X), `PSGUID_STORAGE` needs to be a macro which expands to an initializer rather than being a global variable.

Today's question was really just settling the prerequisites for tomorrow's topic. Stay tuned.

Raymond Chen

**Follow**

