

Re: difference between typedef and #define

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.language/2005-07/msg00015.html>

- *From:* "Doug Harrison [MVP]" <dsh@xxxxxxxx>
 - *Date:* Thu, 30 Jun 2005 19:53:33 -0500
-

On Thu, 30 Jun 2005 18:06:41 -0500, Drew Myers wrote:

> But if I, without using typedef or #define write:
>
> int* i,j,k; // * next to int and not next to i
>
> all of the variables are int*

No, only i is int*. The rest are int.

For background on this, see:

Is ``int* p;" right or is ``int *p;" right?
http://www.research.att.com/~bs/bs_faq2.html#whitespace

> whereas:
>
> int *i,j,k; // * next to i and not next to int
>
> produces the results you explain. Why does typedef/#define
> behave differently than typing it out as I did above?

Macros do pure textual substitution. So, given:

```
#define T int*  
T i, j, k;
```

The preprocessor turns this into:

```
int* i, j, k;
```

The typedef mechanism is different. It doesn't do textual substitution but instead creates synonyms for types. So, given:

```
typedef int* T;  
T i, j, k;
```

This results in all the variables having the type T, which is int*. To see another difference, consider:

Re: difference between typedef and #define

```
const T x;  
T const x;
```

The const (or volatile) applies to the variable x in the first case as well as the second, not to the referent (i.e. *x), and that's another difference between the macro T and typedef T. So, when you write a pointer or reference typedef, you should also provide const/volatile forms as necessary, and your users will have to learn to use LPCSTR instead of "const LPSTR" or "LPSTR const", to use a Windows example.

—
Doug Harrison
Microsoft MVP – Visual C++
.

- **References:**
 - ◆ **[Re: difference between typedef and #define](#)**
 - ◇ From: Drew Myers
- Prev by Date: **[Re: difference between typedef and #define](#)**
- Next by Date: **[RE: IGlobalInterfaceTable \(was: Debugging com ptr t... in JNI fr](#)**
- Previous by thread: **[Re: difference between typedef and #define](#)**
- Next by thread: **[Re: difference between typedef and #define](#)**
- Index(es):
 - ◆ **[Date](#)**
 - ◆ **[Thread](#)**