

Re: C4239 – why here?

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.vc/2005-09/msg00235.html>

- *From:* "Goran Pusic" <goran_pusic@xxxxxxxxxx>
 - *Date:* Fri, 23 Sep 2005 13:48:01 +0200
-

```
> void f(int& i)
> {
> i = 10;
> }
>
> double d = 0;
> f(d);
> //would you expect d to be 10?
```

Yes, I see. In fact, i would expect the compiler to refuse this outright, it just makes no sense at all! Using intrinsic type promotions on references like this, tsk, tsk, naughty compiler... Pascal doesn't allow this! Does C#? I hope not :-))

But, that's not my situation. I have X as a "polymorphic worker class", and it may or not change inside X, I don't care. I just want to it passed to F to get polymorphic behaviour in F depending on the calling context).

Sort-of:

BaseX

```
{ virtual f() }
```

```
X1:BaseX { overridden virtual F() }
```

X2, X3...

and then F(BaseX&) gets called like this: F(X1(params)), F(X2(params)), F(X3(params)) etc...

Goran.

• *Follow-Ups:*

- ◆ **[Re: C4239 – why here?](#)**

◇ *From:* Tom Widmer [VC++ MVP]

- ◆ **[Re: C4239 – why here?](#)**

◇ *From:* Doug Harrison [MVP]

Re: C4239 – why here?

- **References:**

- ◆ **C4239 – why here?**

- ◇ *From:* Goran Pusic

- ◆ **Re: C4239 – why here?**

- ◇ *From:* Tom Widmer [VC++ MVP]

- Prev by Date: **Re: C4239 – why here?**
- Next by Date: **Re: C4239 – why here?**
- Previous by thread: **Re: C4239 – why here?**
- Next by thread: **Re: C4239 – why here?**
- Index(es):
 - ◆ **Date**
 - ◆ **Thread**