
Subject: Re: The problem with 'Null'

Posted by [gridem](#) on Fri, 20 Mar 2009 07:16:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

Thank you very much for detailed answers! It's very useful for me to understanding intrinsics of Upp.

But let's me to understand my opinion. I think that Null approach is not narrow-minded but generic. It uses for Value, it uses for many simple types.

Because I use a lot the templates, overloaded functions and code generation I have to utilize the generic methods for every types that I want to use. And I use Null as some kind of parameter state that I can treat as:

1. Init/Non init
2. Error on function return.
3. Default values to call the function to distinguish it from nondefault values:

```
void some_fun(int a, int b = Null, bool c = Null, ...)
```

```
{  
    if (b == Null)  
        b = some_complex_calculated_value(a);  
    ...  
}
```

etc

So to see this approach for String's and Vector's I have to do workaround. May be I should not use the generics but another mechanism?
