
Subject: Re: Inull values in arrays

Posted by [Paul Van Delst\[1\]](#) on Thu, 16 Dec 2010 15:14:00 GMT

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

Hello,

Axel M wrote:

> Hi Gray,

>

> I see your point and agree to some extent. But I would also see the

> opposite arguments: what should then be the result of $3 * \text{!NULL}$ or

> $3 / \text{!NULL}$? Do you propose that !NULL is handled as a 0? This might

> often lead to more confusion, and for indexing purposes it is

> definitely not a 0.

In IDL, storing a null in an otherwise homogeneous array makes said array a list. Why not use a list? The OP could still call it an array.... :o)

Purely for perspective into the issue, ruby handles operations with null values thusly:

```
irb(main):001:0> 3 + nil
```

```
TypeError: nil can't be coerced into Fixnum
```

```
  from (irb):1:in `+'
```

```
  from (irb):1
```

and python similarly:

```
>>> 3 + None
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in ?
```

```
TypeError: unsupported operand type(s) for +: 'int' and 'NoneType'
```

cheers,

paulv
