



Quora Contributor

Opinions expressed by Forbes Contributors are their own.

TECH 8/09/2013 @ 12:24PM | 4 958 views

What Is The Worst Mistake Ever Made In Computer Programming?

One very good



Anders Kaseorg, MIT 2008, Co-Founder of Ksplice, Inc.

candidate, according to Tony Hoare (developer of [quicksort](#) and one of the early implementors of ALGOL), is his invention of the **null reference**. By allowing programmers to submit a non-object anywhere an object would be accepted, a safety hole is opened in *every API* that's invisible to the compiler, necessitating extensive manual checks and runtime assertions. In a [2009 presentation](#), Hoare estimated the impact of this quick-and-dirty hack at around a billion dollars:

“ I call it my billion-dollar mistake. It was the invention of the null reference in 1965. At that time, I was designing the first comprehensive type system for references in an object oriented language (ALGOL W). My goal was to ensure that all use of references should be absolutely safe, with checking performed automatically by the compiler. But I couldn't resist the temptation to put in a null reference, simply because it was so easy to implement. This has led to innumerable errors, vulnerabilities, and system crashes, which have probably caused a billion dollars of pain and damage in the last forty years.

The problem is that the null reference has been overloaded to mean at least seven different things:

- a semantic non-value, such as a property that is applicable to some objects but not others;
- a missing value, such as the result of looking up a key that is not present in a key-value store, or the input used to request deletion of a key;
- a sentinel representing the end of a sequence of values;
- a placeholder for an object that has yet to be initialized or has already been destroyed, and therefore cannot be used at this time;
- a shortcut for requesting some default

or previous value;

- an invalid result indicating some exceptional or error condition;
- or, most commonly, just a special case to be tested for and rejected, or to be ignored and passed on where it will cause unpredictable problems somewhere else.

Since the null reference is a legitimate value of every object type as far as the language is concerned, the compiler has no way to distinguish between these different uses. The programmer is left with the responsibility to keep them straight, and it's all too easy to occasionally slip up or forget.

We are finally starting to see mainstream languages take steps toward providing alternatives to null, such as Scala's `Option[]` and C# 4.0's `Nullable<>`. But it will likely be many more decades before APIs are updated to take advantage of them.

*This question [originally appeared on Quora](#).
More questions on [Computer Programming](#):*

- [How does NASA use machine learning?](#)
- [What are some common misconceptions about computer science degrees?](#)
- [What are the best Python scripts you've ever written?](#)

RECOMMENDED BY FORBES

[The Richest Person In Every State](#)

[Why 'Spider-Man: Homecoming 2' Is Marvel's Best Choice To Start Phase Four](#)

This article is available online at: <http://onforb.es/1cQRDgD>

2016 Forbes.com LLC™ All Rights Reserved