

Performance difference between dot notation versus method call in Objective-C

Asked 16 years, 3 months ago Modified 2 years, 8 months ago Viewed 5k times



You can use a standard dot notation or a method call in Objective-C to access a property of an object in Objective-C.

13



```
myObject.property = YES;
```

or



```
[myObject setProperty:YES];
```

Is there a difference in performance (in terms of accessing the property)? Is it just a matter of preference in terms of coding style?

objective-c

performance

Share

Improve this question

Follow

edited Nov 10, 2008 at 15:09



Bill the Lizard

405k ● 211 ● 572 ● 889

asked Aug 29, 2008 at 16:39



Ronnie Liew

18.3k ● 14 ● 47 ● 50

5 Answers

Sorted by: Highest score (default)



Dot notation for property access in Objective-C **is** a message send, just as bracket notation. That is, given this:

21



```
@interface Foo : NSObject
@property BOOL bar;
@end
```



```
Foo *foo = [[Foo alloc] init];
foo.bar = YES;
[foo setBar:YES];
```



The last two lines will compile exactly the same. The only thing that changes this is if a property has a `getter` and/or `setter` attribute specified; however, all it does is change what message gets sent, not whether a message is sent:

```
@interface MyView : NSView
@property(getter=isEmpty) BOOL empty;
@end

if ([someView isEmpty]) { /* ... */ }
if (someView.empty) { /* ... */ }
```

Both of the last two lines will compile identically.

Share Improve this answer Follow

answered Aug 30, 2008 at 3:06



Chris Hanson

55k ● 8 ● 74 ● 104



Check out [article from Cocoa is My Girlfriend](#). The gist of it, is that there is no performance penalty of using one over the other.

5



However, the notation does make it more difficult to see what is happening with your variables and what your variables are.



Share

edited Aug 29, 2008 at 19:09

answered Aug 29, 2008 at 16:45



Improve this answer

Follow



Misha M

11.3k ● 17 ● 55 ● 67



The only time you'll see a performance difference is if you do not mark a property as "nonatomic". Then `@synthesize` will automatically add synchronization code around the setting of your property, keeping it thread safe - but slower to set and access.

5



Thus mostly you probably want to define a property like:

```
@property (nonatomic, retain) NSString *myProp;
```



Personally I find the dot notation generally useful from the standpoint of you not having to think about writing correct setter methods, which is not completely trivial even for nonatomic setters because you must also remember to release the old value properly. Using template code helps but you can always make mistakes and it's generally repetitious code that clutters up classes.

A pattern to be aware of: if you define the setter yourself (instead of letting `@synthesize` create it) and start having other side effects of setting a value you should probably make the setter a normal method instead of calling using the property notation.

Semantically using properties appears to be direct access to the actual value to the caller and anything that varies from that should thus be done via sending a message, not accessing a property (even though they are really both sending messages).

Share

Improve this answer

Follow

edited Apr 18, 2022 at 5:25



a stone arachnid

1,279 ● 1 ● 16 ● 27

answered Sep 15, 2008 at 5:44



Kendall Helmstetter
Gelner

75k ● 26 ● 131 ● 151



As far as I've seen, there isn't a significant performance difference between the two. I'm reasonably certain that in most cases it will be 'compiled' down to the same code.

0



If you're not sure, try writing a test application that does each method a million times or so, all the while timing how long it takes. That's the only way to be certain (although it may vary on different architecture.)



Share Improve this answer Follow

answered Aug 29, 2008 at 16:44



Daniel Jennings

6,490 ● 3 ● 33 ● 44



Also read this blog post on Cocoa with Love:

0



<http://cocoawithlove.com/2008/06/speed-test-nsmanagedobject-objc-20.html>

There the author compares the speed of custom accessor and dot notations for NSObject, and finds no difference. However, KVC access (setValue:forKey:) appears to be about twice as slow.



Share Improve this answer Follow

answered Sep 16, 2008 at 22:19



charles

11.7k ● 3 ● 35 ● 46