

# *More Than Po:* Debugging In lldb

@MicheleTitolo

Lead Software Engineer @ Capital One

**uldb**

- LLDB basics

Advanced thread breakpoints  
Watchpoint debugging

- Debugging with script

*What We'll Cover*



# *lldb Commands*



frame lang == lladb lang

Swift!

# Basics

po



$\text{expr} - 0 \quad \text{--}$

```
(lldb) po self.myCar  
<Car: 0x7fc8206334d0>
```



```
(lldb) po self.canChangeGearTo(gear: Gear.Park)
true
```

–description



**(lldb) po self.myCar**

2014 Tesla S UIDeviceRGBColorSpace 0 0 1 1

p

expr --

```
(lldb) p self.myCar.year  
(NSInteger) $1 = 2014
```



```
(lldb) p *self.myCar
(Car) $1 = {
  NSObject = {
    isa = Car
  }
  _running = NO
  _make = 0x000000010077d088 @"Tesla"
  _model = 0x000000010077d0a8 @"S"
  _year = 2014
  _color = 0x00007fd531eb9e30
  _gear = Park
}
```



```
(lldb) po self.myCar  
<SwiftCars.Car: 0x7fe8bb456a60>
```

```
(lldb) p *self.myCar  
error: <EXPR>:1:1: error: '*' is not a prefix unary operator  
*self.myCar  
^
```

use expr to modify  
values at runtime

```
(lldb) po self.myCar
```

```
2014 Tesla S UIDeviceRGBColorSpace 0 0 1 1
```

```
(lldb) expr self.myCar.year = 2013
```

```
(NSInteger) $0 = 2013
```

```
(lldb) po self.myCar
```

```
2013 Tesla S UIDeviceRGBColorSpace 0 0 1 1
```



frame variable

fr v

```
(lldb) fr v
(ViewController *) self = 0x00007f9628520cd0
(SEL) _cmd = "updateTipLabelsForBillAmount:"
(float) billAmount = 33
(float) tipPercentage = 0.2000000003
(float) tipAmount = 6.59999999
(float) total = 39.59999985
```



```
(lldb) fr v -F self.myCar
self.myCar = 0x00007fe8bb456a60
self.myCar =
self.myCar.isa = SwiftCars.Car
self.myCar.make._core._owner = nil
self.myCar.model._core._owner = nil
self.myCar.color = 0x00007fe8bb452c80
self.myCar.color.isa = UIColorDeviceRGBColor
self.myCar.color._systemColorName = 0x00007fe8bb4424c0
"blueColor"
self.myCar.color.redComponent = 0
self.myCar.color.greenComponent = 0
self.myCar.color.blueComponent = 1
self.myCar.color.alphaComponent = 1
self.myCar.color.cachedColor = 0x00000000000000000000
self.myCar.color.cachedColorOnceToken = 0
```



bt

thread backtrace

(lldb) bt

```
* thread #1: tid = 0x66c2, 0x0000000101a90330 Cars`-[
ViewController viewDidLoad](self=0x00007fc3d2c21570,
_cmd=0x0000000102f21cfb) + 448 at ViewController.m:28, queue =
'com.apple.main-thread', stop reason = breakpoint 1.1
  * frame #0: 0x0000000101a90330 Cars`-[ViewController
viewDidLoad](self=0x00007fc3d2c21570, _cmd=0x0000000102f21cfb) +
448 at ViewController.m:28
    frame #1: 0x000000010284f090 UIKit`-[UIViewController
loadViewIfNeeded] + 738
    frame #2: 0x000000010284f28e UIKit`-[UIViewController view] +
27
    frame #3: 0x000000010276b5e9 UIKit`-[UIWindow
addRootViewControllerViewIfPossible] + 58
    frame #4: 0x000000010276b9af UIKit`-[UIWindow
_setHidden:forced:] + 247
```

step / s

▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔄

↔

📄

📄

📄

📁 Cars > Cars > ViewController.m > No Selection

Cars

PID 3668, Paused

🔍

⏸

CPU

0%

Memory

17.8 MB

Disk

Zero KB/s

Network

Zero KB/s

Thread 1

Queue: com.apple.main-thread (serial)

0 -[ViewController viewDidLoad]

1 -[UIViewController loadViewIfRe...

2 -[UIViewController view]

3 -[UIWindow addRootViewContr...

4 -[UIWindow \_setHidden:forced:]

5 -[UIWindow makeKeyAndVisible]

6 -[UIApplication \_callInitialization...

7 -[UIApplication \_runWithMainSc...

8 -[UIApplication workspaceDidE...

9 \_\_31-[FBSSerialQueue perform...

10 \_\_CFRUNLOOP\_IS\_CALLING\_...

11 \_\_CFRunLoopDoBlocks

12 \_\_CFRunLoopRun

13 CFRunLoopRunSpecific

14 -[UIApplication \_run]

15 UIApplicationMain

16 main

17 start

▶ Thread 2

Queue: com.apple.libdispatch-mana...

▶ Thread 3

Queue: FBSSerialQueue (serial)

▶ Thread 4

▶ Thread 5

19

20 - (void)viewDidLoad {

21 [super viewDidLoad];

22 // Do any additional setup after loading the view,

23 typically from a nib.

24 self.myCar = [[Car alloc] init];

25 self.myCar.make = @"Tesla";

26 self.myCar.model = @"S";

27 self.myCar.year = 2014;

28 self.myCar.color = [UIColor blueColor];

29 [self.myCar honk];

30

Thread 1: breakpoint 1.1

🔍

▶

🏠

⬇

⬆

🔍

Cars > Thread 1 > 0 -[ViewController viewDidLoad]

(lldb)

🗑

📄

📄

All Output ↕

🗑

📄

📄

▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔍

↺

📄

📄

📄

📁 Cars

🔍

📄 Cars > Cars > ViewController.m

No Selection

▼ Cars

PID 3668, Paused

⏸

CPU

0%

Memory

17.8 MB

Disk

Zero KB/s

Network

Zero KB/s

Thread 1

Queue: com.apple.main-thread (serial)

0 -[ViewController setMyCar:]

1 -[ViewController viewDidLoad]

2 -[UIViewController loadViewIfRe...

3 -[UIViewController view]

4 -[UIWindow addRootViewContr...

5 -[UIWindow \_setHidden:forced:]

6 -[UIWindow makeKeyAndVisible]

7 -[UIApplication \_callInitialization...

8 -[UIApplication \_runWithMainSc...

9 -[UIApplication workspaceDidE...

10 \_\_31-[FBSSerialQueue perform...

11 \_\_CFRunLoop\_IS\_CALLING\_...

12 \_\_CFRunLoopDoBlocks

13 \_\_CFRunLoopRun

14 CFRunLoopRunSpecific

15 -[UIApplication \_run]

16 UIApplicationMain

17 main

18 start

▶ Thread 2

Queue: com.apple.libdispatch-mana...

▶ Thread 3

Queue: FBSSerialQueue (serial)

▶ Thread 4

▶ Thread 5

7 //

8

9 #import "ViewController.h"

10 #import "Car.h"

11

12 @interface ViewController ()

13 @property(n nonatomic, strong) Car \*myCar;

14 @property (weak, nonatomic) IBOutlet UILabel \*speedLabel;

15

16 @end

17

18 @implementation ViewController

19

🔍

▶

🏠

⬇

⬆

📄

🔍

Cars > Thread 1 > 0 -[ViewController setMyCar:]

(lldb) s

(lldb) |

🗑

📄

📄

All Output ↕



next / n



▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔄

↔

📄

📄

📄

📁

🔍

⚠️

📏

🗨️

🔧

◀

▶

Cars > Cars > ViewController.m > No Selection

Cars

PID 3668, Paused

🔍

⏸️

CPU

0%

Memory

17.8 MB

Disk

Zero KB/s

Network

Zero KB/s

Thread 1

Queue: com.apple.main-thread (serial)

0 -[ViewController viewDidLoad]

1 -[UIViewController loadViewIfRe...

2 -[UIViewController view]

3 -[UIWindow addRootViewContr...

4 -[UIWindow \_setHidden:forced:]

5 -[UIWindow makeKeyAndVisible]

6 -[UIApplication \_callInitialization...

7 -[UIApplication \_runWithMainSc...

8 -[UIApplication workspaceDidE...

9 \_\_31-[FBSSerialQueue perform...

10 \_\_CFRunLoop\_IS\_CALLING\_...

11 \_\_CFRunLoopDoBlocks

12 \_\_CFRunLoopRun

13 CFRunLoopRunSpecific

14 -[UIApplication \_run]

15 UIApplicationMain

16 main

17 start

▶ Thread 2

Queue: com.apple.libdispatch-mana...

▶ Thread 3

Queue: FBSSerialQueue (serial)

19

20 - (void)viewDidLoad {

21 [super viewDidLoad];

22 // Do any additional setup after loading the view,

23 typically from a nib.

24 self.myCar = [[Car alloc] init];

25 self.myCar.make = @"Tesla";

26 self.myCar.model = @"S";

27 self.myCar.year = 2014;

28 self.myCar.color = [UIColor blueColor];

29 [self.myCar honk];

30

Thread 1: step over

🔍

▶

⏸️

🏠

⬇️

⬆️

📄

🔍

Cars > Thread 1 > 0 -[ViewController viewDidLoad]

(lldb) n

(lldb)

🔧

📄

📄

🗑️

All Output ↕

🗑️

📄

📄

thread step-out /  
finish





▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔄

↔

📱

📺

🖨

📁 Cars > Cars > ViewController.m > No Selection

Cars

PID 3732, Paused

⏸

CPU

0%

Memory

17.9 MB

Disk

Zero KB/s

Network

Zero KB/s

Thread 1

Queue: com.apple.main-thread (serial)

0 -[ViewController viewDidLoad]

1 -[UIViewController loadViewIfRe...

2 -[UIViewController view]

3 -[UIWindow addRootViewContr...

4 -[UIWindow \_setHidden:forced:]

5 -[UIWindow makeKeyAndVisible]

6 -[UIApplication \_callInitialization...

7 -[UIApplication \_runWithMainSc...

8 -[UIApplication workspaceDidE...

9 \_\_31-[FBSSerialQueue perform...

10 \_\_CFRUNLOOP\_IS\_CALLING\_...

11 \_\_CFRunLoopDoBlocks

12 \_\_CFRunLoopRun

13 CFRunLoopRunSpecific

14 -[UIApplication \_run]

15 UIApplicationMain

16 main

17 start

Thread 2

Queue: com.apple.libdispatch-mana...

Thread 5

Queue: FBSSerialQueue (serial)

23

24

25

26

27

28

29

30

31

32

```
self.myCar = [[Car alloc] init];
self.myCar.make = @"Tesla";
self.myCar.model = @"S";
self.myCar.year = 2014;
self.myCar.color = [UIColor blueColor];
[self.myCar honk];

[self.myCar addObserver:self forKeyPath:@"speed" options:
    NSKeyValueObservingOptionNew context:nil];
[NSTimer scheduledTimerWithTimeInterval:15 target:self
    selector:@selector(createRandomCarAndSpeedUp)
    userInfo:nil repeats:NO];
```

Thread 1: step out

⏏

▶

🏠

⬇

⬆

🔍

Cars > Thread 1 > 0 -[ViewController viewDidLoad]

(lldb) s

(lldb) finish

2015-03-05 10:15:48.849 Cars[3732:205901] HONK HONK

(lldb) |

🔍

📄

📱

🖨

⌂

All Output ↕

🗑

📄

📱

continue / c

▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔍

↺

📄

📄

📄

📁 Cars > Cars > ViewController.m > No Selection

📁 Cars  
PID 3668, Running

CPU 0%

Memory 18.4 MB

Disk Zero KB/s

Network Zero KB/s

19

20 - (void)viewDidLoad {

21 [super viewDidLoad];

22 // Do any additional setup after loading the view,

23 typically from a nib.

24 self.myCar = [[Car alloc] init];

25 self.myCar.make = @"Tesla";

26 self.myCar.model = @"S";

27 self.myCar.year = 2014;

28 self.myCar.color = [UIColor blueColor];

29 [self.myCar honk];

30

📄 Cars

(lldb) n

(lldb) c

Process 3668 resuming

2015-03-05 10:10:59.847 Cars[3668:202485] HONK HONK

🔍

📄

📄

📄

All Output ↕

🗑️

📄

📄



~/lldbinit

```
settings set prompt [llldb]$
```

```
command alias bd breakpoint disable
```

```
command alias be breakpoint enable
```

```
command alias bdel breakpoint delete
```

```
command alias bcommand breakpoint command add
```

```
command alias commands breakpoint command list
```

process

**(lldb) process attach -p 5416**

Process 5416 stopped

\* thread #1: tid = 0xe4b02, 0x0000000106e234de

libsystem\_kernel.dylib`mach\_msg\_trap + 10, queue = 'com.apple.main  
thread', stop reason = signal SIGSTOP

frame #0: 0x0000000106e234de libsystem\_kernel.dylib`mach\_msg\_trap +  
10

libsystem\_kernel.dylib`mach\_msg\_trap:

-> 0x106e234de <+10>: retq

0x106e234df <+11>: nop

libsystem\_kernel.dylib`mach\_msg\_overwrite\_trap:

0x106e234e0 <+0>: movq %rcx, %r10

0x106e234e3 <+3>: movl \$0x1000020, %eax

Executable module set to "/Users/michele/Library/Developer/CoreSimulator/Devices/  
5304657F-1710-42A1-87CA-E5358E87F4C8/data/Containers/Bundle/Application/312BA401-3B1E-4A17-  
B7E8-BAAAE13EC463/Cars.app/Cars".

Architecture set to: x86\_64h-apple-ios.

**(lldb) c**

Process 5416 resuming



help

help breakpoint

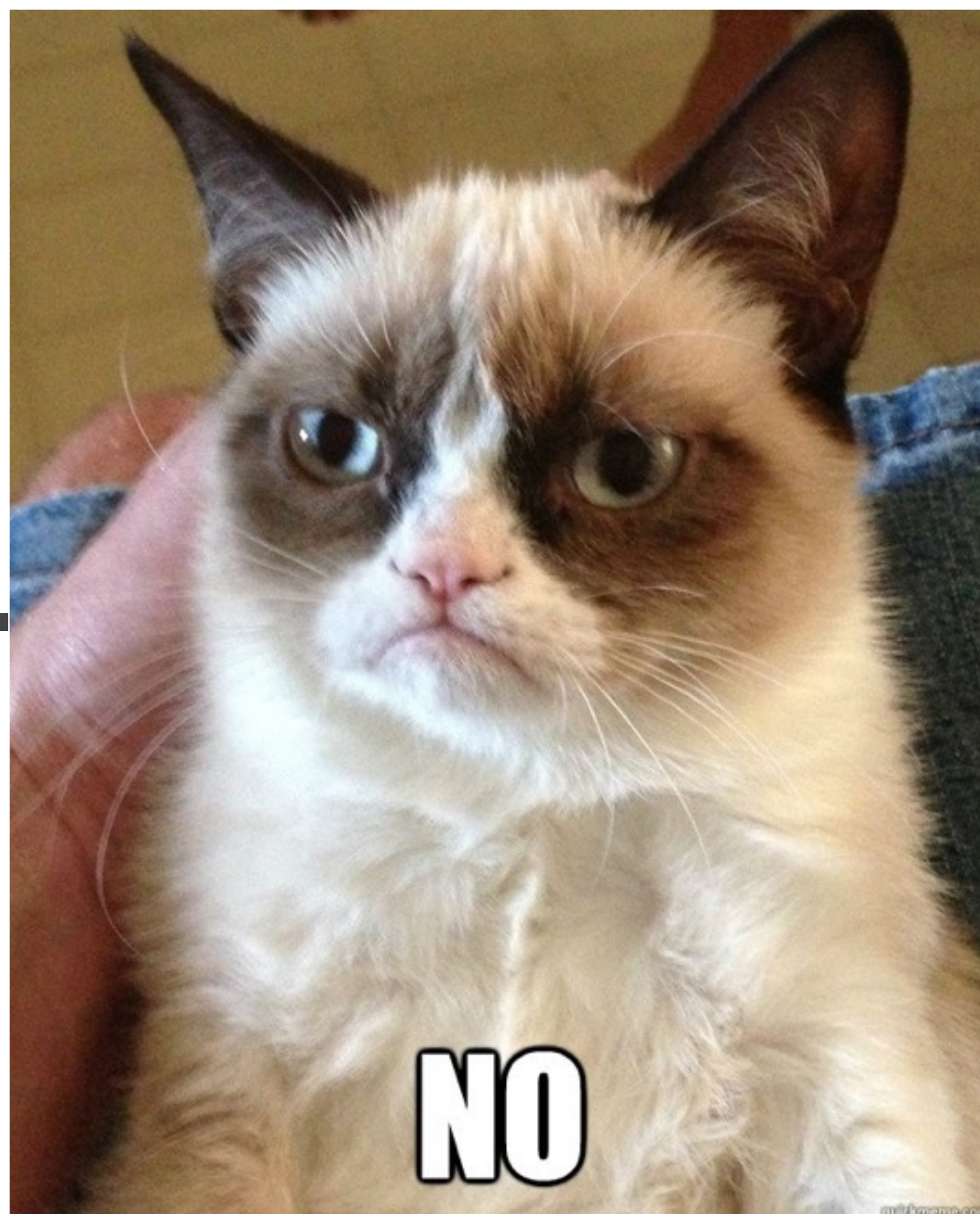
help breakpoint  
command



help breakpoint  
command add

—h

—



# Thread

thread info

**(lldb) thread info**

```
thread #1: tid = 0x66c2, 0x0000000101a90330 Cars`-[ViewController  
viewDidLoad](self=0x00007fc3d2c21570, _cmd=0x0000000102f21cfb) +  
448 at ViewController.m:28, queue = 'com.apple.main-thread', stop  
reason = breakpoint 1.1
```

thread list

## (lldb) thread list

Process 921 stopped

```
* thread #1: tid = 0x66c2, 0x0000000101a90330 Cars`-[ViewController viewDidLoad](self=0x00007fc3d2c21570,
_cmd=0x0000000102f21cfb) + 448 at ViewController.m:28, queue =
'com.apple.main-thread', stop reason = breakpoint 1.1
  thread #2: tid = 0x66da, 0x0000000104bcc232
libsystem_kernel.dylib`kevent64 + 10, queue =
'com.apple.libdispatch-manager'
  thread #3: tid = 0x66dc, 0x0000000104bc651a
libsystem_kernel.dylib`semaphore_wait_trap + 10, queue =
'FBSSerialQueue'
  thread #4: tid = 0x66dd, 0x0000000104bcb94a
libsystem_kernel.dylib`__workq_kernreturn + 10
  thread #5: tid = 0x66de, 0x0000000104bcb94a
libsystem_kernel.dylib`__workq_kernreturn + 10
```

thread until  
<linenum>



**(lldb) thread until 31**  
Process 1615 resuming  
**(lldb)**

thread return  
<expr>

```
(lldb) thread return NO  
(lldb) thread return [NSNumber numberWithInt:7]  
(lldb) thread return @"Hello"
```



```
(lldb) thread return true
```

```
error: Error returning from frame 0 of thread 1: We only support  
setting simple integer and float return types at present..
```

# Breakpoints



breakpoint list

br list



**(lldb) br list**

Current breakpoints:

1: file = 'Cars/Car.m', line = 50, locations = 1, resolved = 1,  
hit count = 0

1.1: where = Cars`-[Car changeGearTo:] + 52 at Car.m:50,  
address = 0x00000001026fd4e4, resolved, hit count = 0

2: file = 'Cars/ViewController.m', line = 31, locations = 1,  
resolved = 1, hit count = 1

2.1: where = Cars`-[ViewController viewDidLoad] + 512 at  
ViewController.m:31, address = 0x00000001026fcb40, resolved, hit  
count = 1

br set

br modify

br delete

**(lldb) br set -f ViewController.m -l 31**

Breakpoint 2: where = Cars`-[ViewController viewDidLoad] +  
512 at ViewController.m:31, address = 0x000000010e2bdb40

**(lldb) br set -F "--[Car changeGearTo:]"**

Breakpoint 3: where = Cars`--[Car changeGearTo:] + 20 at  
Car.m:47, address = 0x00000001073ba4c4



**(lldb) br set -F "SwiftCars.Car.changeGearTo(gear: Cars.Gear) -> ()"**

Breakpoint 10: where = SwiftCars`SwiftCars.Car.changeGearTo(gear: SwiftCars.Gear) -> () + 18 at ViewController.swift:35, address = 0x000000010c97e032

Module Name

Function signature

**SwiftCars.Car.changeGearTo(gear: SwiftCars.Gear) -> ()**

Type

Namespaced

Built-in types are in the Swift module



**SwiftCars.Car.increaseSpeedTo(speed: Swift.Int) -> ()**

$$b r \approx b$$



**(lldb) b ViewController.m:31**

**(lldb) br ViewController.m:31**

error: command 'breakpoint' did not recognize 'ViewController .m:31' as valid (subcommand might be invalid).

**(lldb) b Car.m:63**

Breakpoint 2: where = Cars`-[Car increaseSpeedTo:] + 118 at  
Car.m:63, address = 0x0000000100b7a576

**(lldb) br modify -c "speed==15" 2**

**(lldb) c**

Process 3093 resuming

**(lldb)**

▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔍

↺

↻

📄

📄

📄

📁 Cars

🔍

📄 Cars > Cars > Car.m > @implementation Car

📁 Cars

PID 3093, Paused

🔍

⏸

CPU

0%

Memory

18.6 MB

Disk

Zero KB/s

Network

Zero KB/s

Thread 1

Queue: com.apple.main-thread (serial)

0 -[Car increaseSpeedTo:]

1 -[ViewController speedChanged:]

2 -[UIApplication sendAction:to:fr...

3 -[UIControl \_sendActionsForEve...

4 -[UIStepper \_updateCount:]

5 -[UIStepper endTrackingWithTo...

6 -[UIControl touchesEnded:withE...

7 -[UIWindow \_sendTouchesForE...

8 -[UIWindow sendEvent:]

9 -[UIApplication sendEvent:]

10 \_UIApplicationHandleEventFro...

11 \_UIApplicationHandleEventQueue

12 \_\_CFRunLoop\_IS\_CALLING\_...

13 \_\_CFRunLoopDoSources0

14 \_\_CFRunLoopRun

15 CFRunLoopRunSpecific

16 GSEventRunModal

17 UIApplicationMain

18 main

19 start

▶ Thread 2

Queue: com.apple.libdispatch-mana...

▶ Thread 4

▶ Thread 7

▶ Thread 8

54 }

55 }

56 }

57 - (void)increaseSpeedTo:(NSInteger) speed

58 {

59     if (self.gear == Park || self.gear == Reverse || self.gear == Neutral) {

60         return;

61     }

62 }

63 self.speed = speed;

64 }

65 }

66 - (NSString \*)description

67 {

Thread 1: breakpoint 2.1

🔍

▶

⏸

🏠

⬇

⬆

🔍

Cars > Thread 1 > 0 -[Car increaseSpeedTo:]

(lldb) b Car.m:63

Breakpoint 2: where = Cars`-[Car increaseSpeedTo:] + 118 at Car.m:63, address = 0x0000000100b7a576

(lldb) br modify -c "speed==15" 2

(lldb) c

2015-03-05 09:14:22.717 Cars[3093:173119] HONK HONK

Process 3093 resuming

(lldb)

🔍

📄

📄

📄

🔍

All Output ↕

🗑

📄

📄

```
(lldb) br modify -c "speed==15 && self.gear==4" 2
```



```
(lldb) br modify -c "speed==15 && gear==Gear.Drive" 1
```

```
(lldb) br modify -c "speed==20 && [self canChangeGearTo:4]" 2
```

```
(lldb) br modify -c "speed==15 && self.gear==4" -i 17 2
```



**-i <count> ( --ignore-count <count> )**

**Set the number of times this  
breakpoint is skipped before stopping.**

**-i <count> ( --ignore-count <count> )**

**Set the minimum number of times this  
breakpoint is skipped before stopping.**

**(lldb) br delete 2**

1 breakpoints deleted; 0 breakpoint locations disabled.

br enable

br disable

```
(lldb) breakpoint disable 2  
1 breakpoints disabled.
```

```
...  
2: file = 'Cars/ViewController.m', line = 31, locations = 1  
Options: disabled
```

```
(lldb) breakpoint enable 2  
1 breakpoints enabled.
```

```
...  
2: file = 'Cars/ViewController.m', line = 31, locations = 1,  
resolved = 1, hit count = 1
```

br command

```
(lldb) br command add -o "fr v" 2  
(lldb) fr v  
(Car *) self = 0x00007fb59ad142b0  
(SEL) _cmd = "changeGearTo:"  
(Gear) gear = Neutral
```



```
(lldb) br command add 1
```

```
Enter your debugger command(s).  Type 'DONE' to end.
```

```
> fr v
```

```
> continue
```

```
> DONE
```

```
(lldb) fr v
```

```
(Car *) self = 0x00007f8061625490
```

```
(SEL) _cmd = "changeGearTo:"
```

```
(Gear) gear = Reverse
```

```
(lldb) continue
```

```
Process 2068 resuming
```

```
Command #2 'continue' continued the target.
```

**(lldb) br command list 3**

Breakpoint 3:

Breakpoint commands:

fr v

**(lldb) br command delete 3**

Breakpoints Added In  
Console Do Not Persist  
Between Runs

# Watchpoints



*A Watchpoint Tracks  
A Value Over Time*

...Kind Of Like Kvo!



```
[self.myCar addObserver:self
                  forKeyPath:@"speed"
                  options:NSKeyValueObservingOptionNew
                  context:KV0Context];
```

```
- (void)observeValueForKeyPath:(NSString *)keyPath ofObject:(id)object
change:(NSDictionary *)change context:(void *)context
{
    if (car.speed > 1000) {
        NSLog(@"WHY AM I GOING SO FAST %ld",
              (long)car.speed);
    }
}
```

```
- (void)setSpeed:(NSInteger) speed
{
    _speed = speed;
    if (_speed > 1000) {
        NSLog(@"WHY AM I GOING SO FAST %ld",
              (long)_speed);
    }
}
```

breakpoint: place  
watchpoint: value

watchpoint set

watch set

```
(lldb) watch set var self.speed
```

```
error: "self" is a pointer and . was used to attempt to  
access "speed". Did you mean "self->speed"?
```

```
(lldb) watch set var self->speed
```

```
error: "speed" is not a member of "(Car *const) self"
```

```
(lldb) watch set var _speed
```



```
(lldb) watch set var self.speed
```

```
Watchpoint created: Watchpoint 1: addr = 0x6080000a4698 size = 8  
state = enabled type = w  
    declare @ '/Projects/SwiftCars/ViewController.swift:28'  
    watchpoint spec = 'self.speed'
```

watch modify -c ...  
-i ...



**(lldb) watch set var \_speed**

Watchpoint created: Watchpoint 1: addr = 0x7fb9c145d768 size  
= 8 state = enabled type = w  
    watchpoint spec = '\_speed'  
    new value: 0

**(lldb) watch modify -c '(\_speed==15)'**

**(lldb) c**

Process 2150 resuming

Watchpoint 1 hit:

old value: 0

new value: 15

**(lldb)**



```
(lldb) watch set var self.speed
```

```
Watchpoint created: Watchpoint 1: addr = 0x6080000a4698 size = 8  
state = enabled type = w
```

```
declare @ '/Projects/SwiftCars/ViewController.swift:28'
```

```
watchpoint spec = 'self.speed'
```

```
(lldb) watch modify -c 'self.speed==15)' 1
```

```
(lldb) c
```

```
Watchpoint 1 hit:
```

```
(lldb) po self.speed
```

```
6
```

**(lldb) watch command add 1**

Enter your debugger command(s). Type 'DONE' to end.

> p \_speed

> continue

> DONE

**(lldb) c**

Process 2252 resuming

(NSInteger) \$16 = 15

Process 2252 resuming

Command #2 'continue' continued the target.

watch delete

watch enable

watch disable

```
(lldb) watch disable 1  
1 watchpoints disabled.  
(lldb) watch enable 1  
1 watchpoints enabled.
```



The background of the image is a dark, deep blue or black sky filled with numerous small, glowing orange and yellow lanterns. These lanterns are scattered throughout the frame, creating a sense of depth and movement. Some lanterns are larger and more prominent, while others are small specks of light. The overall effect is a magical, ethereal atmosphere. The word "Script" is centered in the middle of the image in a white, elegant, cursive script font.

*Script*



write code to debug  
your code



copy + paste



script

## **(lldb) help script**

Pass an expression to the script interpreter for evaluation and return the results. Drop into the interactive interpreter if no expression is given. This command takes 'raw' input (no need to quote stuff).

Syntax: script [<script-expression-for-evaluation>]

**(lldb) script**

Python Interactive Interpreter. To exit, type 'quit()', 'exit()'.

**>>> print lldb.debugger**

Debugger (instance: "debugger\_47", id: 47)

**>>> print lldb.target**

Cars

**>>> print lldb.process**

SBProcess: pid = 6425, state = stopped, threads = 5, executable =  
Cars

**>>> print lldb.thread**

SBThread: tid = 0x3813e

**>>> print lldb.frame**

frame #0: 0x0000000103075979 Cars`-[ViewController viewDidLoad]  
(self=0x00007f8173f23680, \_cmd=0x0000000104507cfb) + 57 at  
ViewController.m:24

**>>> quit()**

Writing functions

current stack frame

breakpoint useless session dict



```
def breakpoint_func(frame, bp_loc, dict):  
    # implementation here  
    # return false to skip
```

# Sample Case



only stop when  
–changeGearTo:  
is in the call stack

```
find_in_stack = ['-[Car changeGearTo:]']
def continue_ignored(frame, bp_loc, dict):
    global find_in_stack
    names = set([frame.GetFunctionName() for frame
        in frame.GetThread()])
    all_ignored = set(find_in_stack)
    ignored_here = all_ignored.intersection(names)
    if len(ignored_here) == 0:
        return False

quit()
```

stack symbol

function declaration

```
find_in_stack = ['-[Car changeGearTo:]']  
def continue_ignored(frame, bp_loc, dict):
```

```
    global find_in_stack
```

global accessor

```
    names = set([frame.GetFunctionName() for frame  
                 in frame.GetThread()])
```

all functions in stack

```
    all_ignored = set(find_in_stack)
```

```
    ignored_here = all_ignored.intersection(names)
```

```
    if len(ignored_here) == 0:
```

```
        return False
```

check if symbol we want  
is in this stack

```
quit()
```

if it isn't, continue

```
find_in_stack = ['-[Car changeGearTo:]']  
def continue_ignored(frame, bp_loc, dict):  
    global find_in_stack  
    names = set([frame.GetFunctionName() for frame  
        in frame.GetThread()])  
    all_ignored = set(find_in_stack)  
    ignored_here = all_ignored.intersection(names)  
    if len(ignored_here) == 0:  
        return False  
  
quit()
```



Module



```
find_in_stack = ['SwiftCars.Car.changeGearTo']
```

br command add -s  
python #

**(lldb) br command add -s python 2**

Enter your Python command(s). Type 'DONE' to end.

```
def function(frame,bp_loc,internal_dict):
    """frame: the SBFrame for the location at which you stopped
        bp_loc: an SBBreakpointLocation for the breakpoint
        location information
        internal_dict: an LLDB support object not to be used"""
    global find_in_stack
    find_in_stack = ['-[Car changeGearTo:]']
    names = set([frame.GetFunctionName() for frame in
frame.GetThread()])
    all_ignored = set(find_in_stack)
    ignored_here = all_ignored.intersection(names)
    if len(ignored_here) == 0:
        return False
```

**DONE**

▶

■

Cars > iPhone 6

Running Cars on iPhone 6

≡

🔍

↺

↻

📄

📄

📄

📁 Cars

PID 6820, Paused

CPU 0%

Memory 18.6 MB

Disk Zero KB/s

Network Zero KB/s

Thread 1

Queue: com.apple.main-thread (serial)

0 -[Car increaseSpeedTo:]

1 -[Car changeGearTo:]

2 -[ViewController gearChanged:]

3 -[UIApplication sendAction:to:fr...

4 -[UIControl \_sendActionsForEve...

5 -[UISegmentedControl \_setSele...

6 -[UISegmentedControl touches...

7 -[UIWindow \_sendTouchesForE...

8 -[UIWindow sendEvent:]

9 -[UIApplication sendEvent:]

10 \_UIApplicationHandleEventFro...

11 \_UIApplicationHandleEventQueue

12 \_\_CFRunLoopIS\_CALLING\_...

13 \_\_CFRunLoopDoSources0

14 \_\_CFRunLoopRun

15 CFRunLoopRunSpecific

16 GSEventRunModal

17 UIApplicationMain

18 main

19 start

Thread 2

Queue: com.apple.libdispatch-mana...

Thread 4

Thread 6

Thread 7

58 {

59 if (self.gear == Park || self.gear == Reverse || self.gear == Neutral) {

60 return;

61 }

62 }

Thread 1: breakpoint 2.1

def continue\_ignored(frame, bp\_loc, dict):

global find\_in\_stack

names = set([frame.GetFunctionName() for frame in frame.GetThread()])

all\_ignored = set(find\_in\_stack)

ignored\_here = all\_ignored.intersection(names)

if len(ignored\_here) == 0:

return False

quit()

>>> ... .. >>> (lldb) br command add -F

continue\_ignored 2

(lldb) c

Process 6820 resuming

2015-03-05 11:45:58.900 Cars[6820:252143] HONK HONK

(lldb) |

🗑️

📄

📄

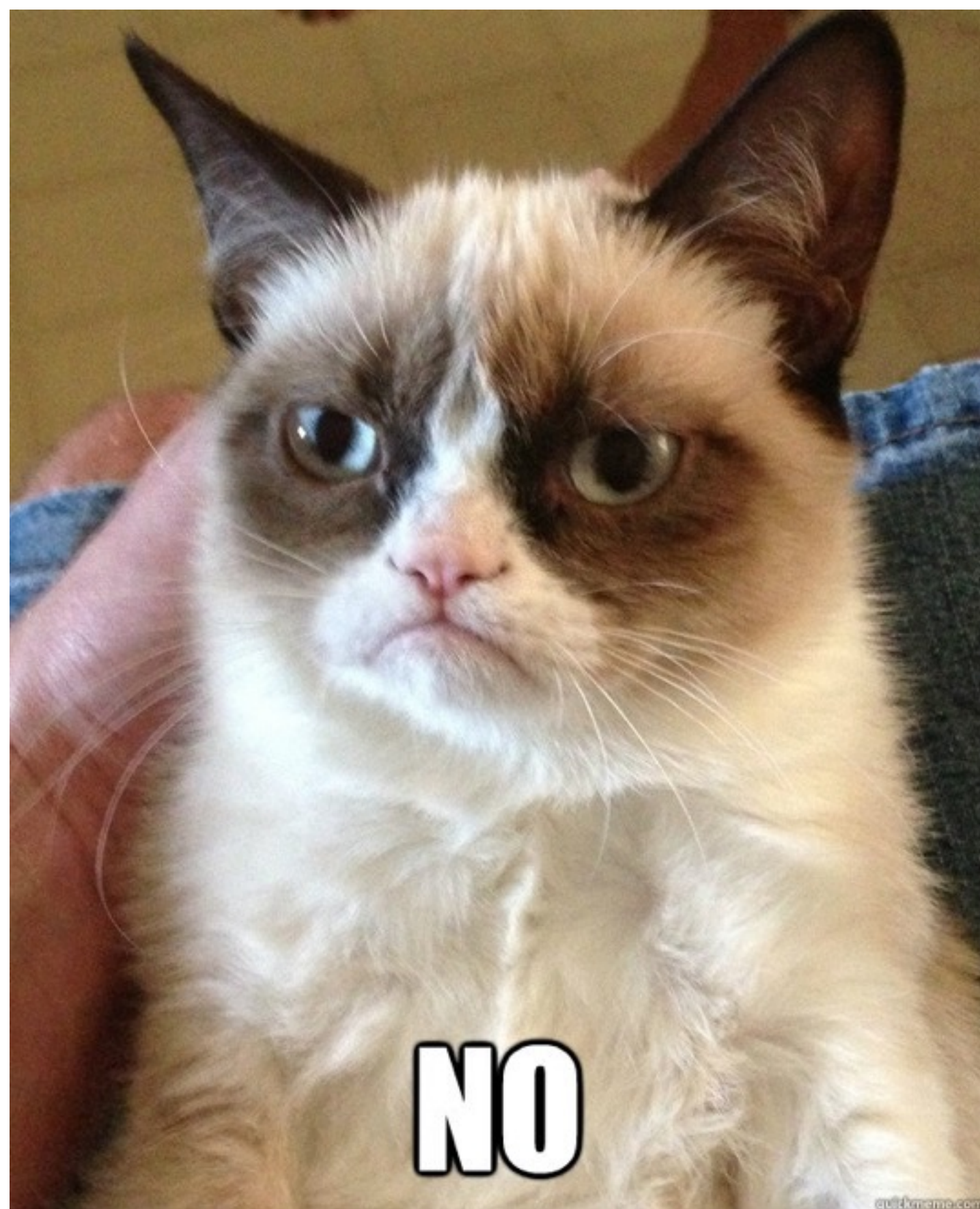
All Output ↕



```
(lldb) br command add -s python 2 -o "print 1+1"
```

type summary

Use A Python  
Function To Better  
Describe A Class



```
def Car_Summary(value,unused):  
    make = value.GetChildMemberWithName("_make")  
    model = value.GetChildMemberWithName("_model")  
    makeSummary = make.GetSummary()  
    modelSummary = model.GetSummary()  
    return makeSummary + " " + modelSummary  
  
quit()
```

value is the frame

get the variables we want

```
def Car_Summary(value, unused):  
    make = value.GetChildMemberWithName("_make")  
    model = value.GetChildMemberWithName("_model")  
    makeSummary = make.GetSummary()  
    modelSummary = model.GetSummary()  
    return makeSummary + " " + modelSummary
```

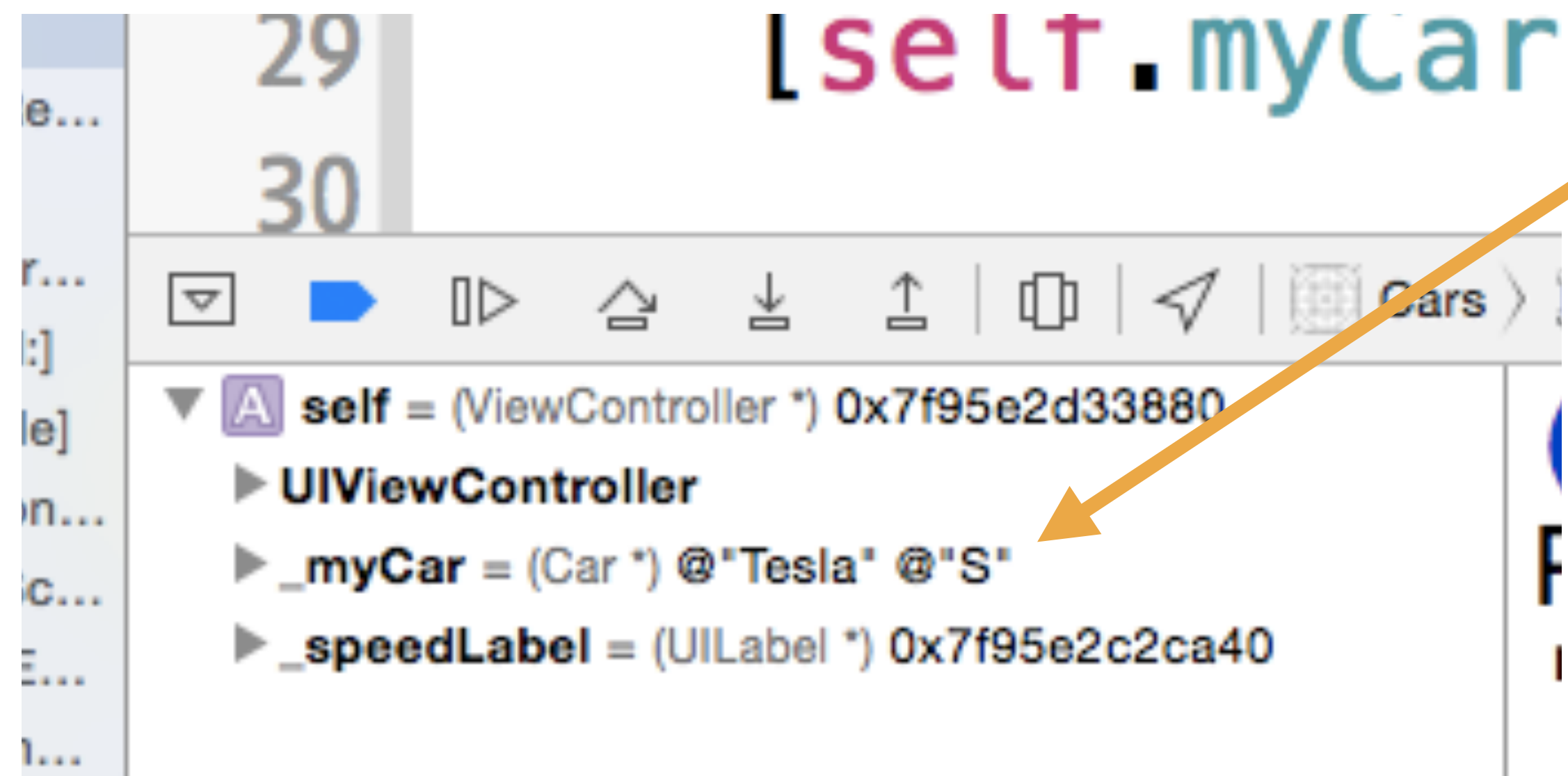
printable  
summaries

quit()

return a string

```
def Car_Summary(value,unused):  
    make = value.GetChildMemberWithName("_make")  
    model = value.GetChildMemberWithName("_model")  
    makeSummary = make.GetSummary()  
    modelSummary = model.GetSummary()  
    return makeSummary + " " + modelSummary  
  
quit()
```







- Only break after another breakpoint has been hit

- Check multiple threads for a symbol
- Data formatters for everything in Objc
- Custom LLDB commands

*More Use Cases*

# *Questions?*

@MicheleTitolo

# Resources

- WWDC '13 Session 413
- WWDC '14 Sessions 409 & 410
- <http://lldb.llvm.org/>
- <http://www.objc.io/issue-19/lldb-debugging.html>
- <http://blog.ittybittyapps.com/blog/2013/11/07/integrating-reveal-without-modifying-your-xcode-project/>
- <https://github.com/facebook/chisel>
- [https://developer.apple.com/library/mac/documentation/IDEs/Conceptual/gdb\\_to\\_lldb\\_transition\\_guide/document/lldb-terminal-workflow-tutorial.html](https://developer.apple.com/library/mac/documentation/IDEs/Conceptual/gdb_to_lldb_transition_guide/document/lldb-terminal-workflow-tutorial.html)

# Photo Credits

- <https://unsplash.com/photos/eMnddgd3pjQ>
- <https://unsplash.com/photos/KpBAYMNf9Tw>