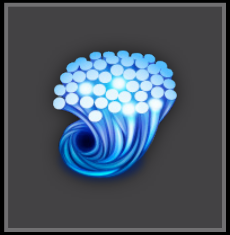
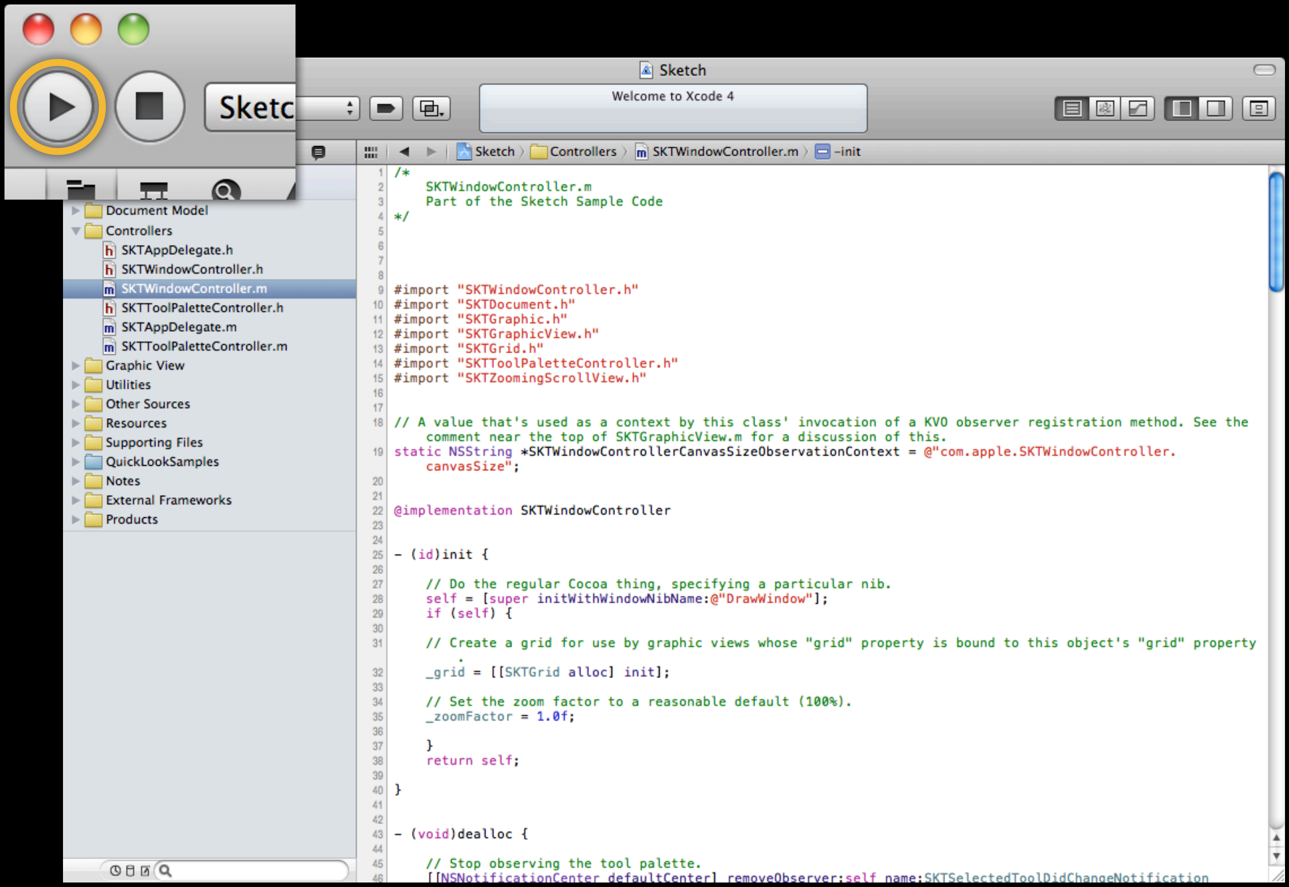
Debugging with Xcode 4 and LLDB

Ken Orr
Xcode Debugger Team

Debugging Is Fundamental

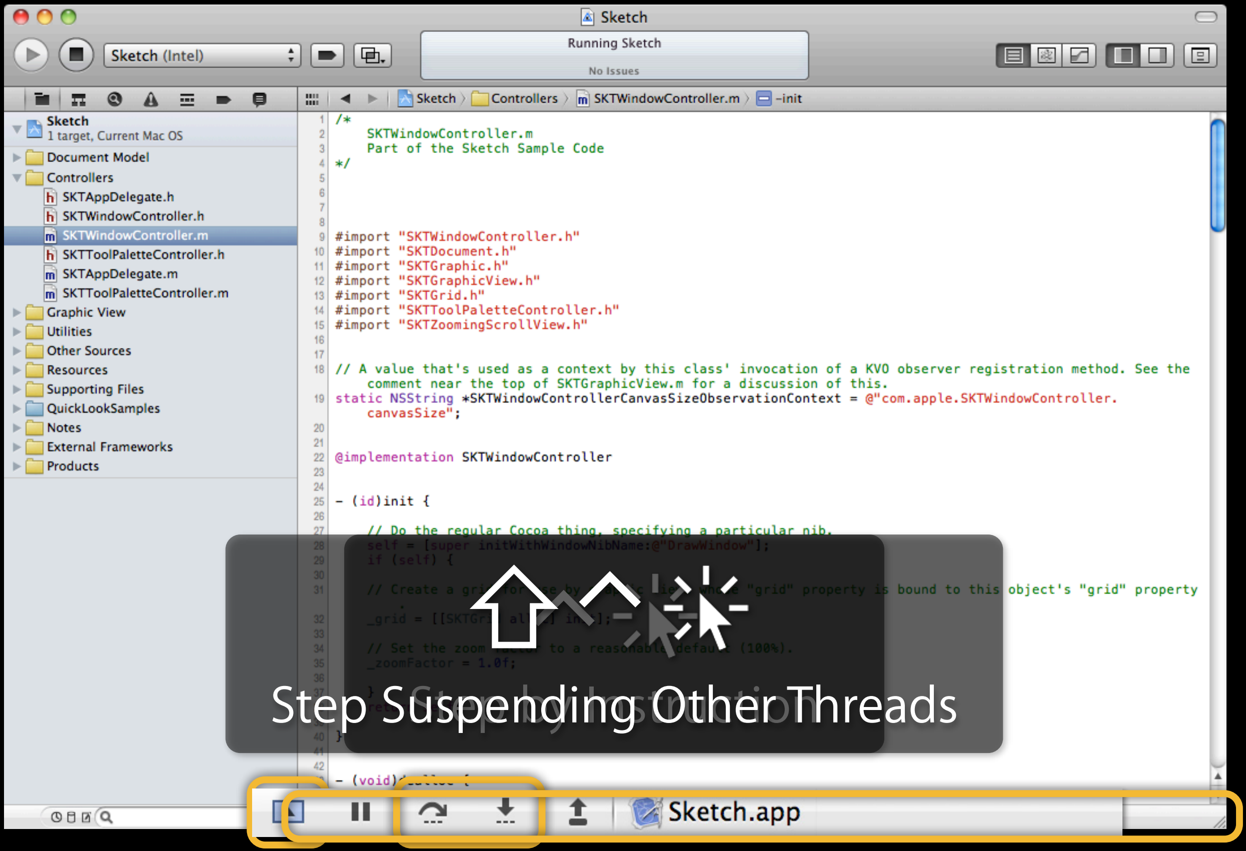
PREVIEW
Color palette
Transparency Effects
Text spacing
Alignment
Grid tool
Writing tools
PROJECT: APPLICATION.APP
ARCHITECT: XCODE
#1





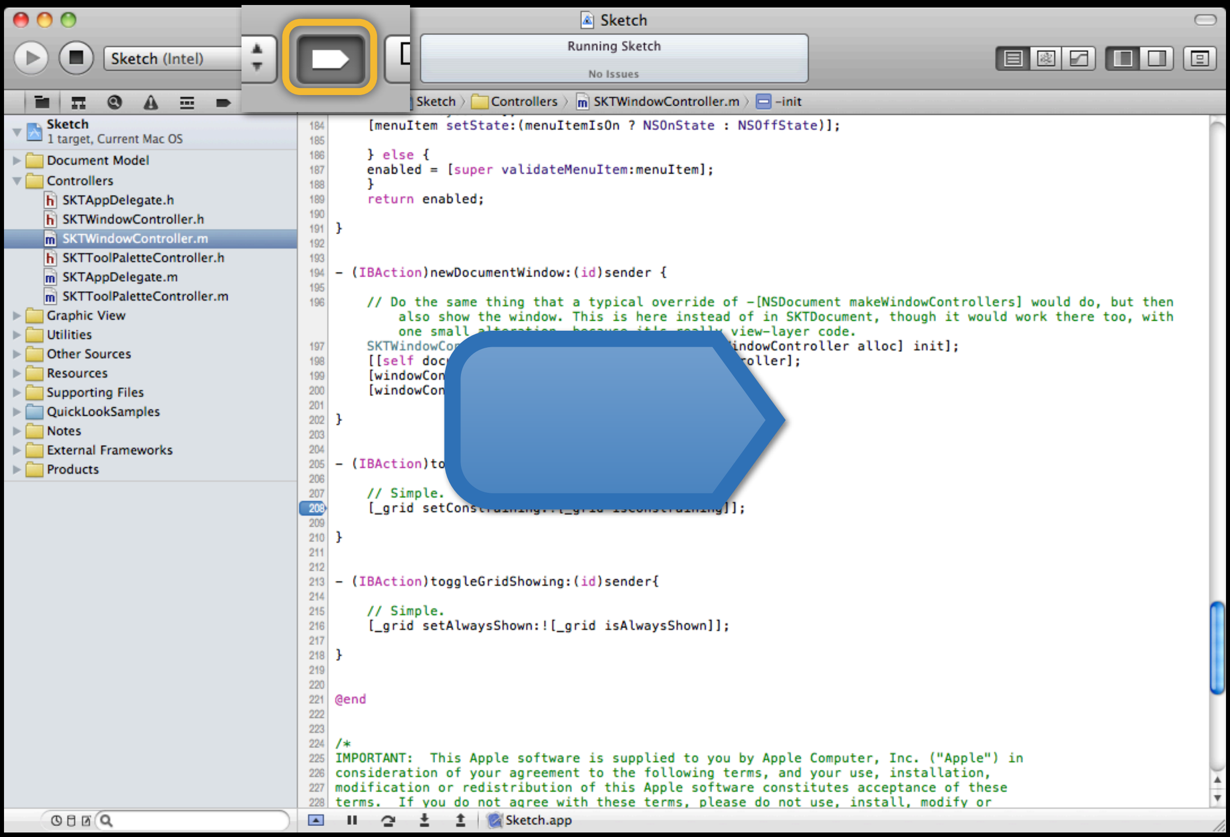
```
/*
    SKTWindowController.m
    Part of the Sketch Sample Code
*/




#import "SKTWindowController.h"
#import "SKTDocument.h"
#import "SKTGraphic.h"
#import "SKTGraphicView.h"
#import "SKTGrid.h"
#import "SKTToolPaletteController.h"
#import "SKTZoomingScrollView.h"


// A value that's used as a context by this class' invocation of a KVO observer registration method. See the
    comment near the top of SKTGraphicView.m for a discussion of this.
static NSString *SKTWindowControllerCanvasSizeObservationContext = @"com.apple.SKTWindowController.
    canvasSize";


@implementation SKTWindowController


- (id)init {

    // Do the regular Cocoa thing, specifying a particular nib.
    self = [super initWithWindowNibName:@"DrawWindow"];
    if (self) {

    // Create a grid for use by graphic views whose "grid" property is bound to this object's "grid" property
        .
    _grid = [[SKTGrid alloc] init];

    // Set the zoom factor to a reasonable default (100%).
    _zoomFactor = 1.0f;

    }
    return self;

}


- (void)dealloc {

    // Stop observing the tool palette.
    [[NSNotificationCenter defaultCenter] removeObserver:self name:SKTSelectedToolDidChangeNotification
```

Xcode 4

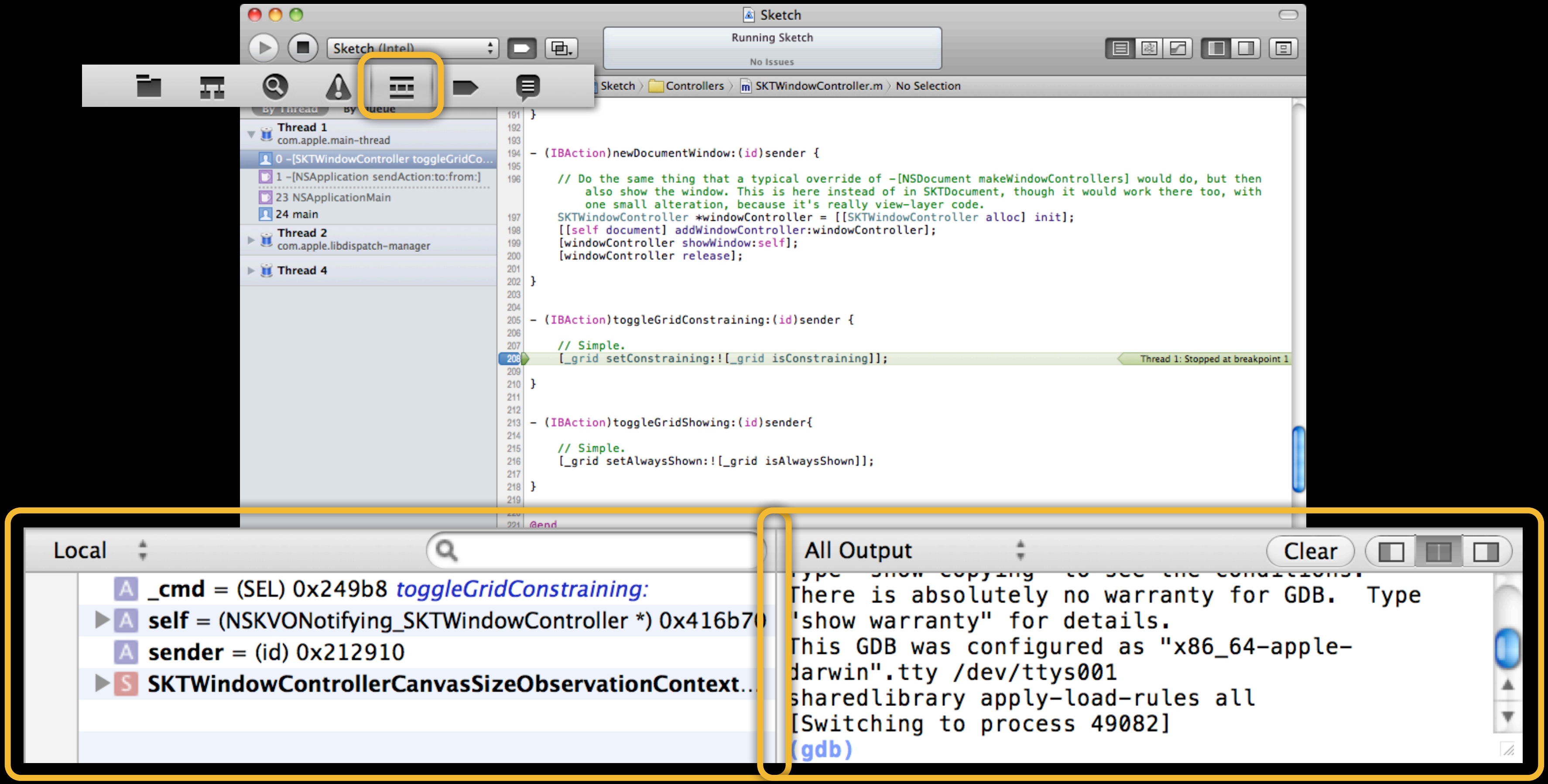


```
/*
    SKTWindowController.m
    Part of the Sketch Sample Code
*/



#import "SKTWindowController.h"
#import "SKTDocument.h"
#import "SKTGraphic.h"
#import "SKTGraphicView.h"
#import "SKTGrid.h"
#import "SKTToolPaletteController.h"
#import "SKTZoomingScrollView.h"


// A value that's used as a context by this class' invocation of a KVO observer registration method. See the
    comment near the top of SKTGraphicView.m for a discussion of this.
static NSString *SKTWindowControllerCanvasSizeObservationContext = @"com.apple.SKTWindowController.
    canvasSize";


@implementation SKTWindowController


- (id)init {

    // Do the regular Cocoa thing, specifying a particular nib.
    self = [super initWithWindowNibName:@"DrawWindow"];
    if (self) {
```

Step Suspending Other Threads



```
    [menuItem setState:(menuItemIsOn ? NSOnState : NSOffState)];

    } else {
    enabled = [super validateMenuItem:menuItem];
    }
    return enabled;

}


- (IBAction)newDocumentWindow:(id)sender {

    // Do the same thing that a typical override of -[NSDocument makeWindowControllers] would do, but then
        also show the window. This is here instead of in SKTDocument, though it would work there too, with
        one small alteration, because it's really view-layer code.
    SKTWindowCo                                     indowController alloc] init];
    [[self doc                                      roller];
    [windowCon
    [windowCon

}


- (IBAction)to

    // Simple.
    [_grid setConstraining:![_grid isConstraining]];

}


- (IBAction)toggleGridShowing:(id)sender{

    // Simple.
    [_grid setAlwaysShown:![_grid isAlwaysShown]];

}


@end


/*
IMPORTANT:  This Apple software is supplied to you by Apple Computer, Inc. ("Apple") in
consideration of your agreement to the following terms, and your use, installation,
modification or redistribution of this Apple software constitutes acceptance of these
terms.  If you do not agree with these terms, please do not use, install, modify or
```

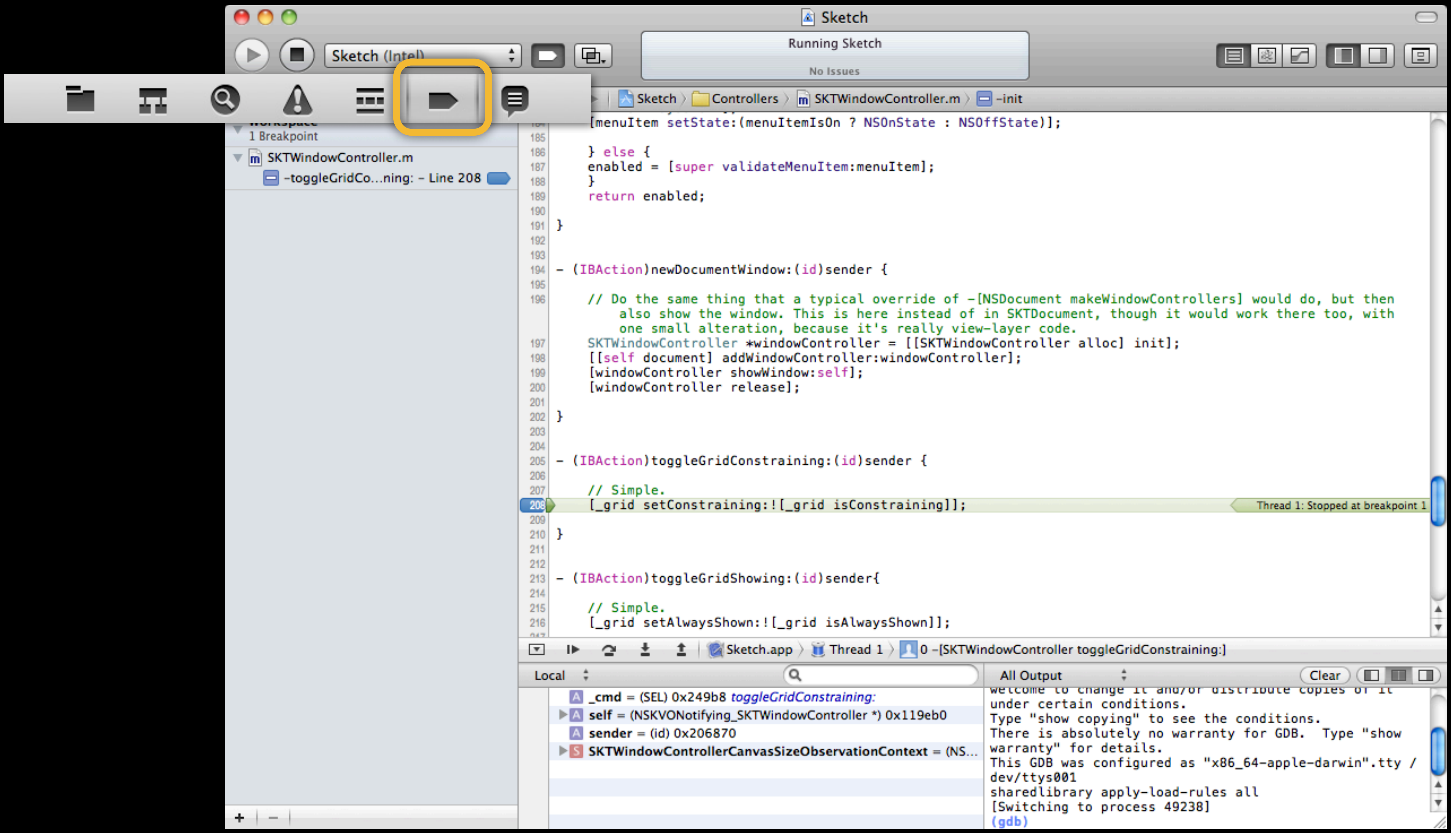


Thread 1
com.apple.main-thread
0 -[SKTWindowController toggleGridCo…
1 -[NSApplication sendAction:to:from:]
23 NSApplicationMain
24 main
Thread 2
com.apple.libdispatch-manager
Thread 4

```
}


- (IBAction)newDocumentWindow:(id)sender {

    // Do the same thing that a typical override of -[NSDocument makeWindowControllers] would do, but then
        also show the window. This is here instead of in SKTDocument, though it would work there too, with
        one small alteration, because it's really view-layer code.
    SKTWindowController *windowController = [[SKTWindowController alloc] init];
    [[self document] addWindowController:windowController];
    [windowController showWindow:self];
    [windowController release];

}


- (IBAction)toggleGridConstraining:(id)sender {

    // Simple.
    [_grid setConstraining:![_grid isConstraining]];

}


- (IBAction)toggleGridShowing:(id)sender{

    // Simple.
    [_grid setAlwaysShown:![_grid isAlwaysShown]];

}


@end
```

```
_cmd = (SEL) 0x249b8 toggleGridConstraining:
self = (NSKVONotifying_SKTWindowController *) 0x416b70
sender = (id) 0x212910
SKTWindowControllerCanvasSizeObservationContext…
```

```
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type
"show warranty" for details.
This GDB was configured as "x86_64-apple-
darwin".tty /dev/ttys001
sharedlibrary apply-load-rules all
[Switching to process 49082]
(gdb)
```



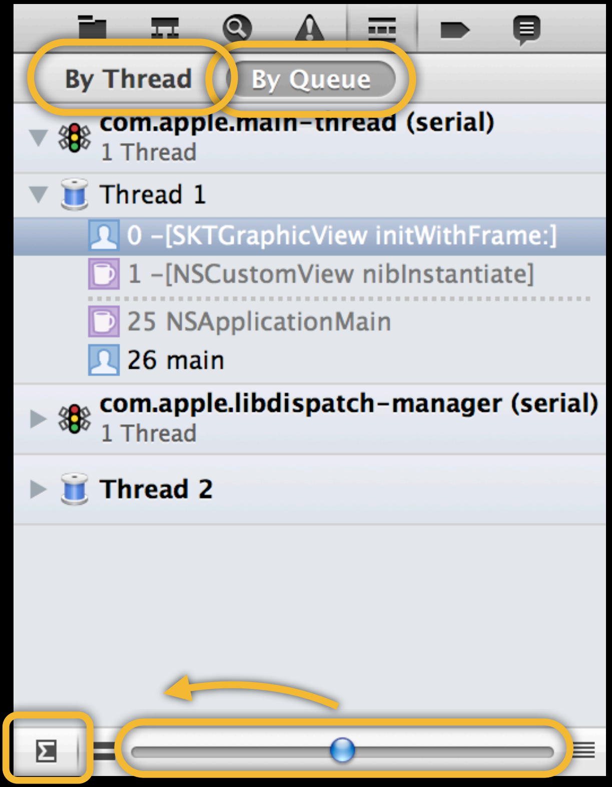
```
        [menuItem setState:(menuItemIsOn ? NSOnState : NSOffState)];

    } else {
    enabled = [super validateMenuItem:menuItem];
    }
    return enabled;

}


- (IBAction)newDocumentWindow:(id)sender {

    // Do the same thing that a typical override of -[NSDocument makeWindowControllers] would do, but then
        also show the window. This is here instead of in SKTDocument, though it would work there too, with
        one small alteration, because it's really view-layer code.
    SKTWindowController *windowController = [[SKTWindowController alloc] init];
    [[self document] addWindowController:windowController];
    [windowController showWindow:self];
    [windowController release];

}


- (IBAction)toggleGridConstraining:(id)sender {

    // Simple.
    [_grid setConstraining:![_grid isConstraining]];

}


- (IBAction)toggleGridShowing:(id)sender{

    // Simple.
    [_grid setAlwaysShown:![_grid isAlwaysShown]];
```

```
_cmd = (SEL) 0x249b8 toggleGridConstraining:
self = (NSKVONotifying_SKTWindowController *) 0x119eb0
sender = (id) 0x206870
SKTWindowControllerCanvasSizeObservationContext = (NS...
```

```
welcome to change it and/or distribute copies of it
under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show
warranty" for details.
This GDB was configured as "x86_64-apple-darwin".tty /
dev/ttys001
sharedlibrary apply-load-rules all
[Switching to process 49238]
(gdb)
```

Debug Navigator

Scope Bar
Group by thread
or by queue.



com.apple.main-thread (serial)
1 Thread

Thread 1

0 -[SKTGraphicView initWithFrame:]

1 -[NSCustomView nibInstantiate]

25 NSApplicationMain

26 main

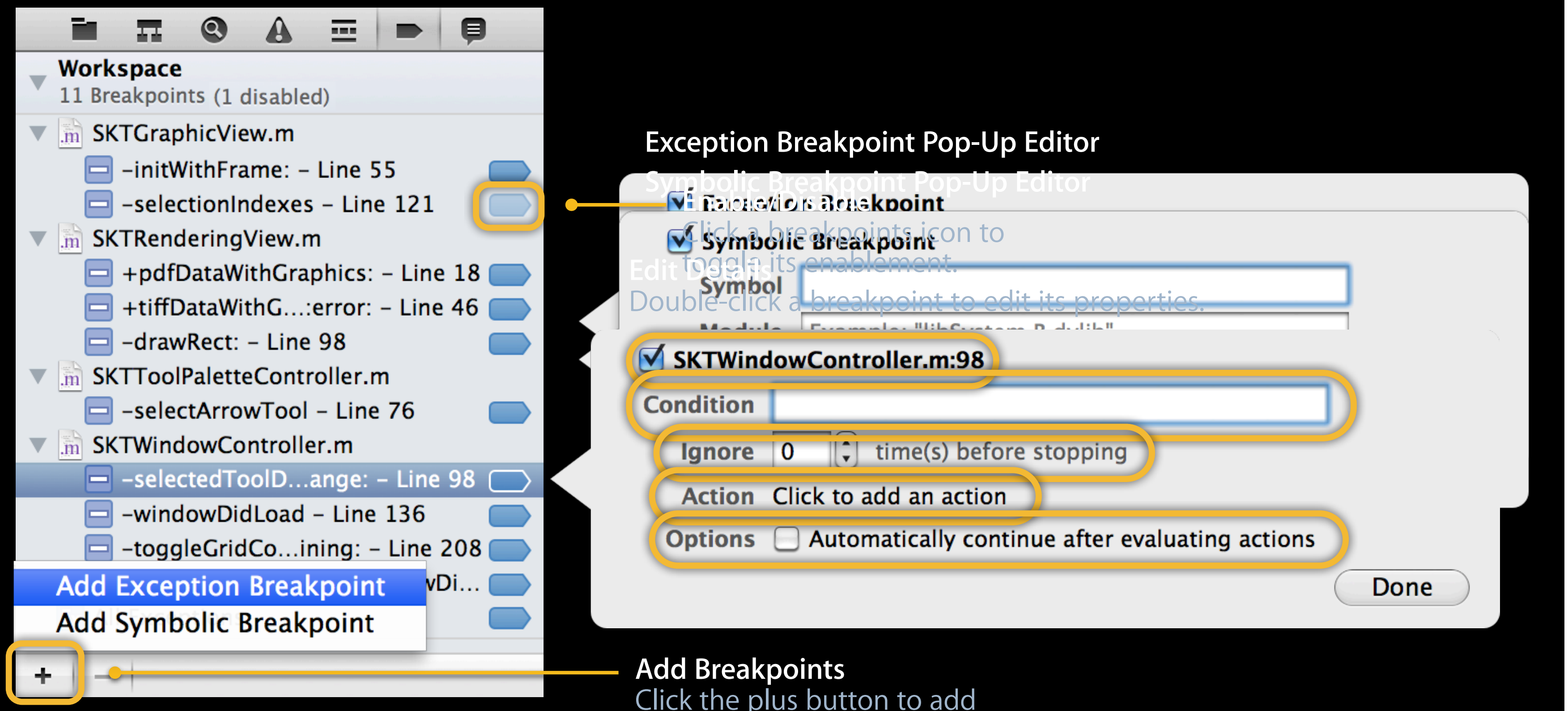
com.apple.libdispatch-manager (serial)
1 Thread

Thread 2

Thread Filter
Filters out threads that
only have symbol-less
stack frames.

Stack Frame Filter
Hide stack frames that
aren’t relevant to what
your debugging.

Breakpoint Navigator



Workspace
11 Breakpoints (1 disabled)

SKTGraphicView.m
- -initWithFrame: – Line 55
- -selectionIndexes – Line 121

SKTRenderingView.m
- +pdfDataWithGraphics: – Line 18
- +tiffDataWithG…:error: – Line 46
- -drawRect: – Line 98

SKTToolPaletteController.m
- -selectArrowTool – Line 76

SKTWindowController.m
- -selectedToolD…ange: – Line 98
- -windowDidLoad – Line 136
- -toggleGridCo…ining: – Line 208
- …wDi…

☑ Exception Breakpoint

☑ Symbolic Breakpoint

Symbol

Module Example: "libSystem.B.dylib"

☑ SKTWindowController.m:98

Condition

Ignore 0 time(s) before stopping

Action Click to add an action

Options □ Automatically continue after evaluating actions

Exception Breakpoint Pop-Up Editor

Symbolic Breakpoint Pop-Up Editor

Enable/Disable
Click a breakpoints icon to toggle its enablement.

Edit Details
Double-click a breakpoint to edit its properties.

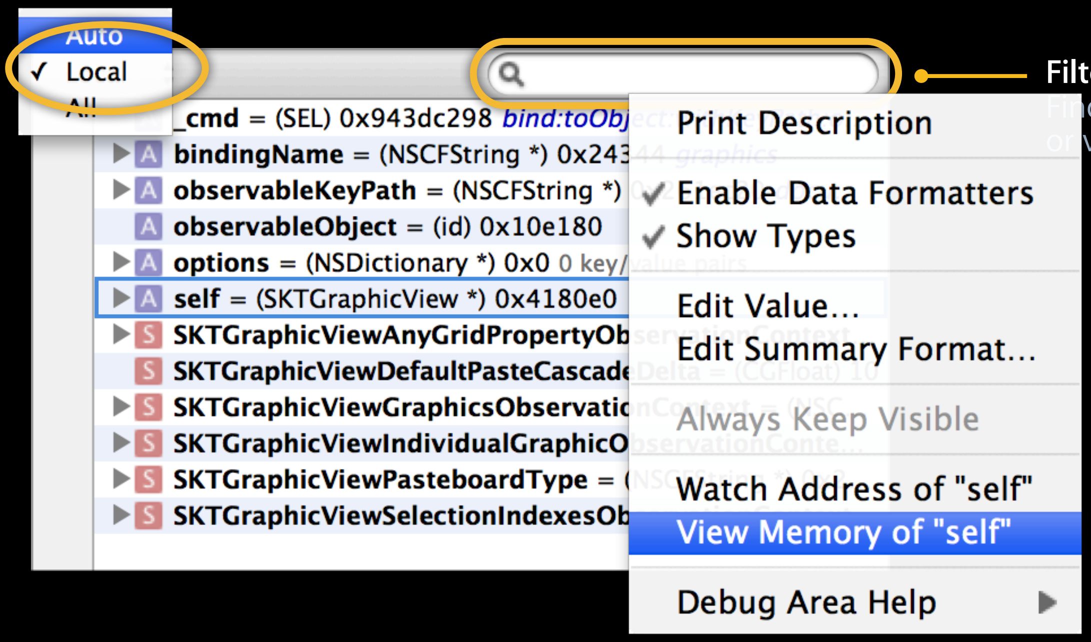
Add Breakpoints
Click the plus button to add symbolic or exception breakpoints.

Demo

Debug and Breakpoint Navigator

Variables View

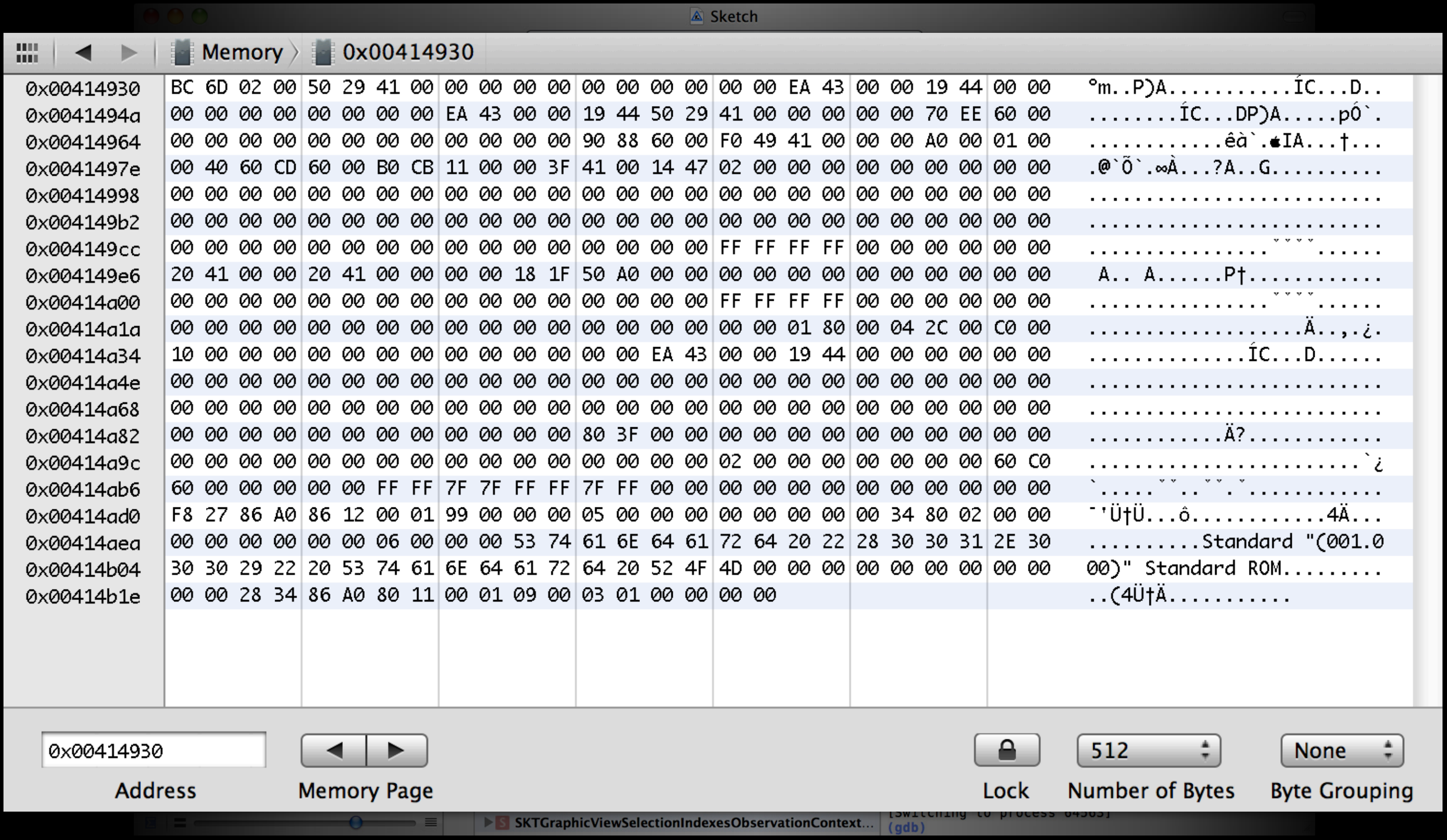
View Mode
The Variables View can automatically figure out what variables are most relevant.



```
_cmd = (SEL) 0x943dc298 bind:toObj
bindingName = (NSCFString *) 0x243
observableKeyPath = (NSCFString *)
observableObject = (id) 0x10e180
options = (NSDictionary *) 0x0 0 key/
self = (SKTGraphicView *) 0x4180e0
SKTGraphicViewAnyGridPropertyOb
SKTGraphicViewDefaultPasteCascade
SKTGraphicViewGraphicsObservatic
SKTGraphicViewIndividualGraphicO
SKTGraphicViewPasteboardType = (
SKTGraphicViewSelectionIndexesOb
```

Filter
Find specific variables or values.

Variables View



```
0x00414930  BC 6D 02 00 50 29 41 00 00 00 00 00 00 00 00 00 00 00 EA 43 00 00 19 44 00 00   °m..P)A...........ÍC...D..
0x0041494a  00 00 00 00 00 00 00 00 EA 43 00 00 19 44 50 29 41 00 00 00 00 00 70 EE 60 00   ........ÍC...DP)A.....pÓ`.
0x00414964  00 00 00 00 00 00 00 00 00 00 00 00 90 88 60 00 F0 49 41 00 00 00 A0 00 01 00   ............êà`.IA...†...
0x0041497e  00 40 60 CD 60 00 B0 CB 11 00 00 3F 41 00 14 47 02 00 00 00 00 00 00 00 00 00   .@`Õ`.∞À...?A..G..........
0x00414998  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ..........................
0x004149b2  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ..........................
0x004149cc  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF 00 00 00 00 00 00   ................ˇˇˇˇ......
0x004149e6  20 41 00 00 20 41 00 00 00 00 18 1F 50 A0 00 00 00 00 00 00 00 00 00 00 00 00    A.. A......P†............
0x00414a00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF FF FF 00 00 00 00 00 00   ................ˇˇˇˇ......
0x00414a1a  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 80 00 04 2C 00 C0 00   ...................Ä..,.¿.
0x00414a34  10 00 00 00 00 00 00 00 00 00 00 00 00 00 EA 43 00 00 19 44 00 00 00 00 00 00   ..............ÍC...D......
0x00414a4e  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ..........................
0x00414a68  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ..........................
0x00414a82  00 00 00 00 00 00 00 00 00 00 00 00 80 3F 00 00 00 00 00 00 00 00 00 00 00 00   ............Ä?............
0x00414a9c  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 02 00 00 00 00 00 00 00 60 C0   ........................`¿
0x00414ab6  60 00 00 00 00 00 FF FF 7F 7F FF FF 7F FF 00 00 00 00 00 00 00 00 00 00 00 00   `.....ˇˇ..ˇˇ.ˇ............
0x00414ad0  F8 27 86 A0 86 12 00 01 99 00 00 00 05 00 00 00 00 00 00 00 00 34 80 02 00 00   ¯'Ü†Ü...ô............4Ä...
0x00414aea  00 00 00 00 00 00 06 00 00 00 53 74 61 6E 64 61 72 64 20 22 28 30 30 31 2E 30   ..........Standard "(001.0
0x00414b04  30 30 29 22 20 53 74 61 6E 64 61 72 64 20 52 4F 4D 00 00 00 00 00 00 00 00 00   00)" Standard ROM.........
0x00414b1e  00 00 28 34 86 A0 80 11 00 01 09 00 03 01 00 00 00 00                           ..(4Ü†Ä...........
```

Address: 0x00414930

Number of Bytes: 512

Byte Grouping: None

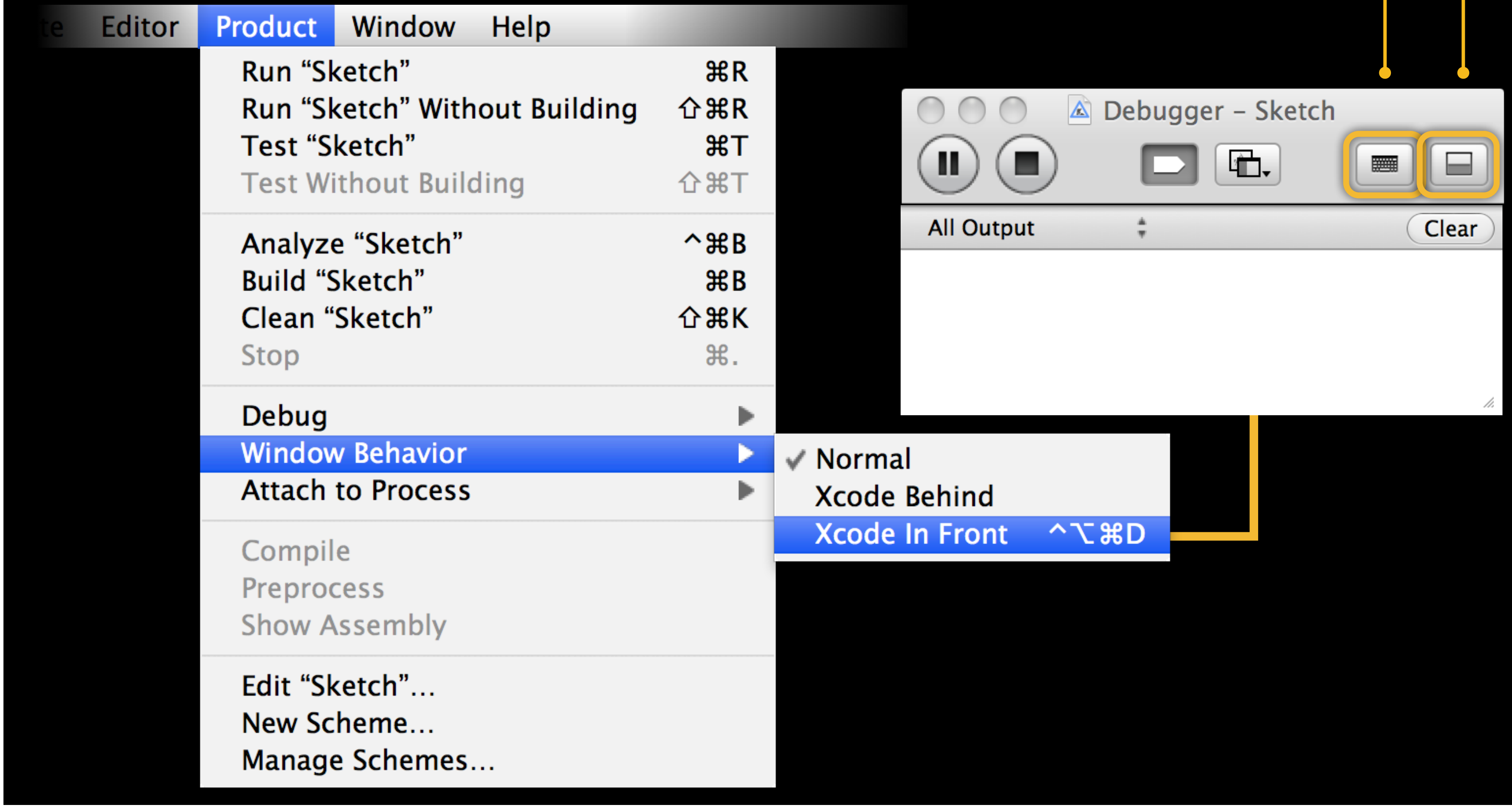
Demo

Variables View

Floating Debugger

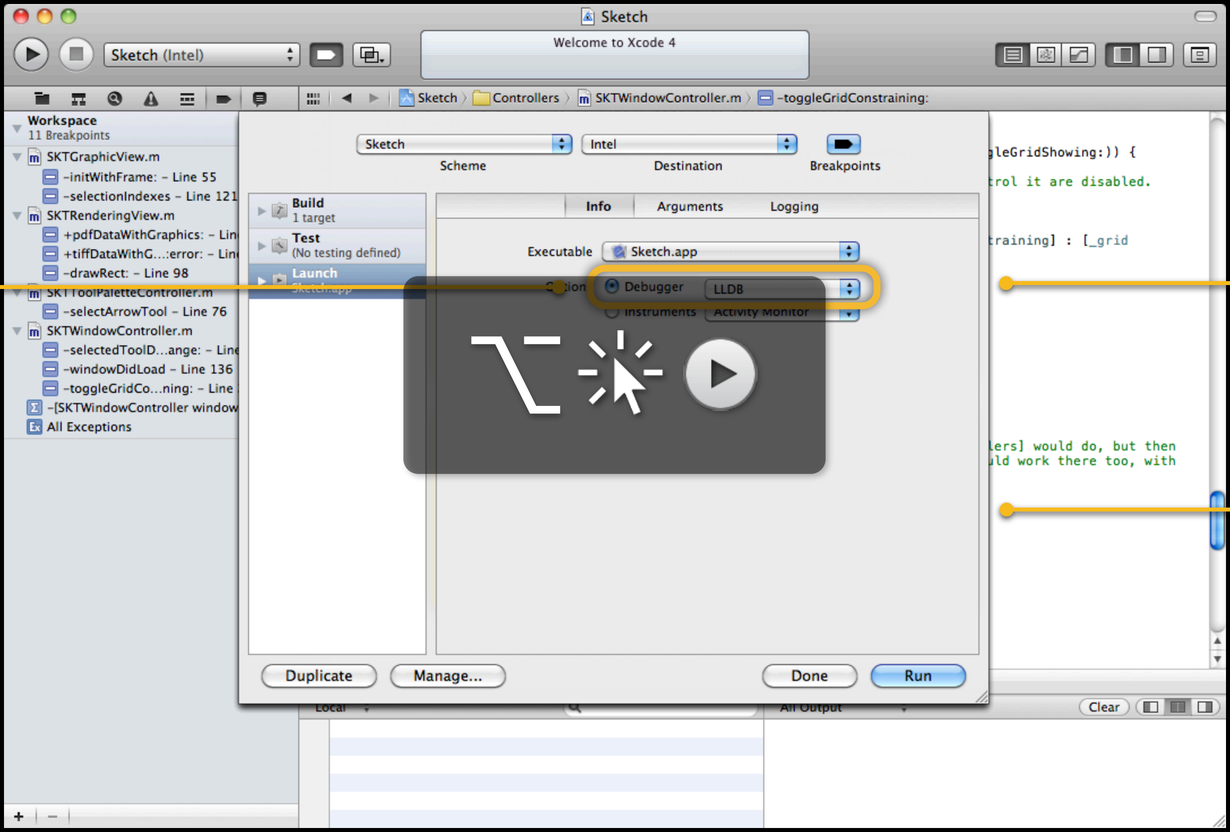
Console Input
Type console commands without the full UI.

Keyboard Input
Prevent unwanted keyboard input.



Schemes

Debugger
Use the new LLDB debugger.


Sketch
Sketch (Intel)
Welcome to Xcode 4
Sketch 〉 Controllers 〉 SKTWindowController.m 〉 -toggleGridConstraining:
Workspace
11 Breakpoints
SKTGraphicView.m
-initWithFrame: - Line 55
-selectionIndexes - Line 121
SKTRenderingView.m
+pdfDataWithGraphics: - Lin
+tiffDataWithG...:error: - Lin
-drawRect: - Line 98
SKTToolPaletteController.m
-selectArrowTool - Line 76
SKTWindowController.m
-selectedToolD...ange: - Line
-windowDidLoad - Line 136
-toggleGridCo...ning: - Line
-[SKTWindowController window
All Exceptions
Build
1 target
Test
(No testing defined)
Launch
Sketch.app
Info
Arguments
Logging
Duplicate
Manage...
Done
Run
Clear
All Output
Local

Scheme: Sketch
Destination: Intel
Breakpoints

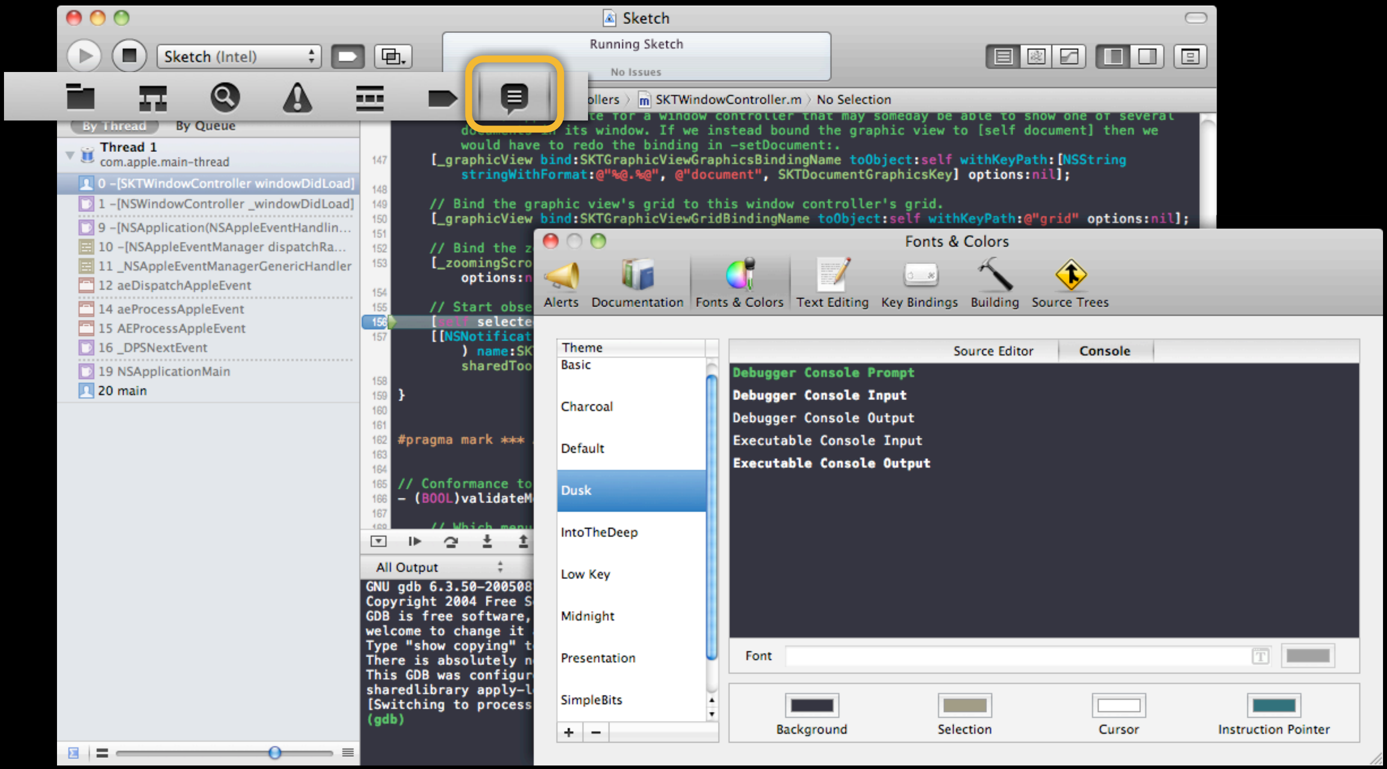
Executable: Sketch.app
Debugger: LLDB
Instruments: Activity Monitor

```
gleGridShowing:)) {
trol it are disabled.
training] : [_grid
lers] would do, but then
ld work there too, with
```

Arguments
Add arguments to pass to your app.

Environment Variables
Add environment variables to pass to your app.

Other Goodies



Thread 1
com.apple.main-thread
0 -[SKTWindowController windowDidLoad]
1 -[NSWindowController _windowDidLoad]
9 -[NSApplication(NSAppleEventHandlin…
10 -[NSAppleEventManager dispatchRa…
11 _NSAppleEventManagerGenericHandler
12 aeDispatchAppleEvent
14 aeProcessAppleEvent
15 AEProcessAppleEvent
16 _DPSNextEvent
19 NSApplicationMain
20 main

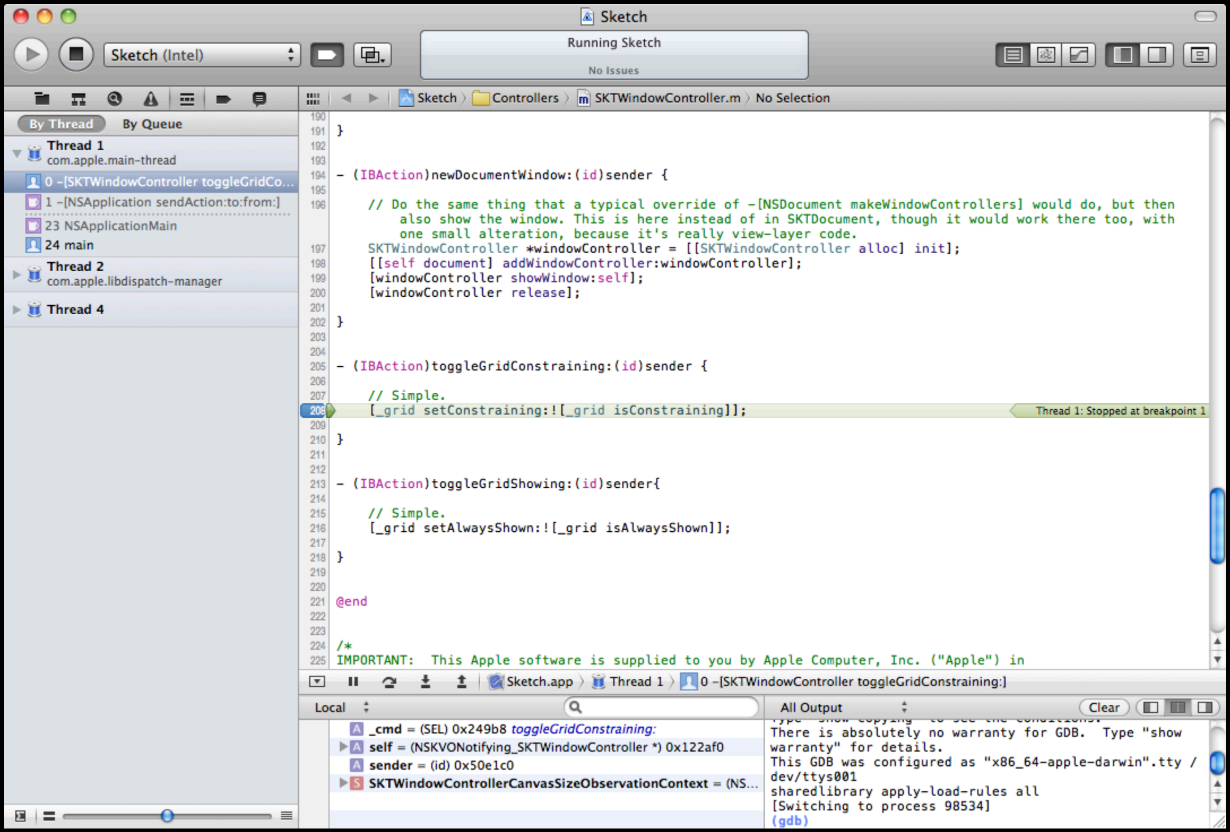
```
    documents in its window. If we instead bound the graphic view to [self document] then we
    would have to redo the binding in -setDocument:.
[_graphicView bind:SKTGraphicViewGraphicsBindingName toObject:self withKeyPath:[NSString
    stringWithFormat:@"%@.%@", @"document", SKTDocumentGraphicsKey] options:nil];

// Bind the graphic view's grid to this window controller's grid.
[_graphicView bind:SKTGraphicViewGridBindingName toObject:self withKeyPath:@"grid" options:nil];
```

```
GNU gdb 6.3.50-200508
Copyright 2004 Free S
GDB is free software,
welcome to change it
Type "show copying" t
There is absolutely n
This GDB was configur
sharedlibrary apply-l
[Switching to process
(gdb)
```

Theme
Basic
Charcoal
Default
Dusk
IntoTheDeep
Low Key
Midnight
Presentation
SimpleBits

Debugger Console Prompt
Debugger Console Input
Debugger Console Output
Executable Console Input
Executable Console Output

DebuggDeAlertsLogsPreferences


Sketch
Sketch (Intel)
Running Sketch
No Issues
By Thread
By Queue
Thread 1
com.apple.main-thread
0 -[SKTWindowController toggleGridCo...
1 -[NSApplication sendAction:to:from:]
23 NSApplicationMain
24 main
Thread 2
com.apple.libdispatch-manager
Thread 4
Sketch › Controllers › SKTWindowController.m › No Selection

```
}


- (IBAction)newDocumentWindow:(id)sender {

    // Do the same thing that a typical override of -[NSDocument makeWindowControllers] would do, but then
        also show the window. This is here instead of in SKTDocument, though it would work there too, with
        one small alteration, because it's really view-layer code.
    SKTWindowController *windowController = [[SKTWindowController alloc] init];
    [[self document] addWindowController:windowController];
    [windowController showWindow:self];
    [windowController release];

}


- (IBAction)toggleGridConstraining:(id)sender {

    // Simple.
    [_grid setConstraining:![_grid isConstraining]];

}


- (IBAction)toggleGridShowing:(id)sender{

    // Simple.
    [_grid setAlwaysShown:![_grid isAlwaysShown]];

}


@end


/*
IMPORTANT:  This Apple software is supplied to you by Apple Computer, Inc. ("Apple") in
```

Thread 1: Stopped at breakpoint 1
Sketch.app › Thread 1 › 0 -[SKTWindowController toggleGridConstraining:]
Local
All Output
Clear

```
_cmd = (SEL) 0x249b8 toggleGridConstraining:
self = (NSKVONotifying_SKTWindowController *) 0x122af0
sender = (id) 0x50e1c0
SKTWindowControllerCanvasSizeObservationContext = (NS...
```

```
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show
warranty" for details.
This GDB was configured as "x86_64-apple-darwin".tty /
dev/ttys001
sharedlibrary apply-load-rules all
[Switching to process 98534]
(gdb)
```

Xcode 4

https://developer.apple.com/wwdc/news/monday.html

LLDB

Modular Debugging Infrastructure

Greg Clayton
Senior LLDB Engineer

LLDB Session Overview

Why?

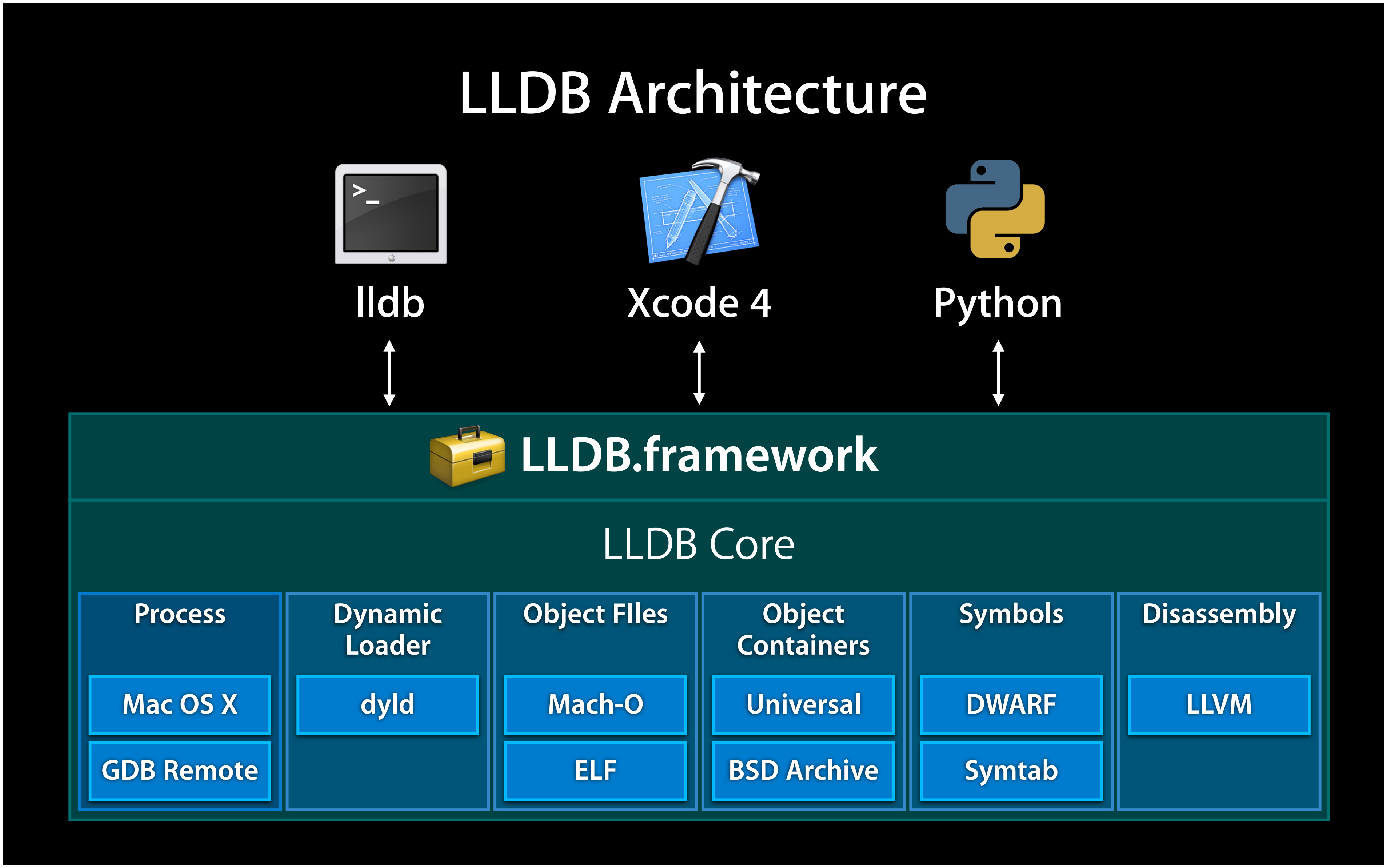
What?

Demo

Conclusion

Why LLDB?

- Speed
- Efficiency
- Accuracy
- Extensibility
- Reusability



LLDB Architecture

lldb

Xcode 4

Python

LLDB.framework

LLDB Core

Process
- Mac OS X
- GDB Remote

Dynamic Loader
- dyld

Object FIles
- Mach-O
- ELF

Object Containers
- Universal
- BSD Archive

Symbols
- DWARF
- Symtab

Disassembly
- LLVM

Performance

Speed

- Information management
- Take advantage of multiple cores

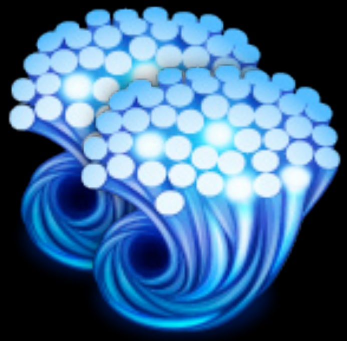
Efficiency

- Memory footprint
- Smarter object file and symbol parsing
  - Don’t ask, don’t parse
  - Fine granularity

Multithreaded Debugging

Better at showing multiple threads

- Per-thread state
- Per-thread runtime control
  - Suspend and resume
  - Control actions
  - Control actions are stackable



Compiler Integration

- Most debuggers invent data structures
  - Functions
  - Types
  - Variables
- Most debuggers have their own expression parser
- Compilers are better at expression parsing

LLVM Integration

- LLDB uses Clang types
- Improved expression fidelity
- Better language support
- Accurate error reporting
- Get new compiler features for free
- Just In Time (JIT) compilation

Scriptability

Python support

- Full access to the LLDB API

Python access

- Command line
- Debugger console
- Breakpoint commands

```
% setenv PYTHONPATH /Xcode4/Library/PrivateFrameworks/
LLDB.framework/Resources/Python
% python
>>> import lldb
```

```
% /Xcode4/usr/bin/lldb
(lldb) script
>>> import lldb
```

```
(lldb) breakpoint set -name main
(lldb) breakpoint command add --script 1
Enter your Python command(s). Type 'DONE' to end.
>>> import lldb
```

Breakpoint Resolution

- Breakpoints are specified symbolically
  - File and line
  - Function by name or regular expression
- Breakpoint can resolve to multiple locations
  - Resolution never stops
  - Locations can individually be enable/disabled
- Breakpoint scopes can be controlled
- Limit breakpoint to one or more shared libraries

Consistent Command Syntax

Easy to learn

- Noun + Verb
- Options
- Autocomplete

Comprehensive help

- help
- apropos

```
(lldb) breakpoint set --file main.c --line 12
```

LLDB Command Line Tool

```
% /Xcode4/usr/bin/lldb --file /tmp/a.out

(lldb) breakpoint set --name main

Breakpoint created: 1 Breakpoint by name: 'main'
with 1 location;

(lldb) run

Launching '/tmp/a.out'  (x86_64)

Process 29428 Stopped

...
```

Demo

Sean Callanan

Conclusion

- Developer preview available with Xcode 4
- lldb comand line debugger
- lldb python module in LLDB.framework
- Open source now
- http://lldb.llvm.org/

More Information

Documentation

Mac OS X Human Interface Guidelines
http://developer.apple.com/ue

Apple Developer Forums

http://devforums.apple.com

Q&A

WWDC10

[redacted]