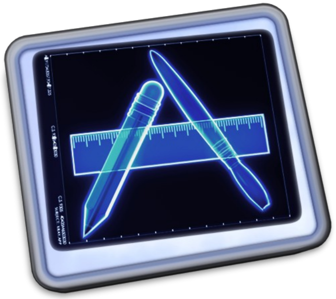


Instruments

Software Profiling – SoSe 2013

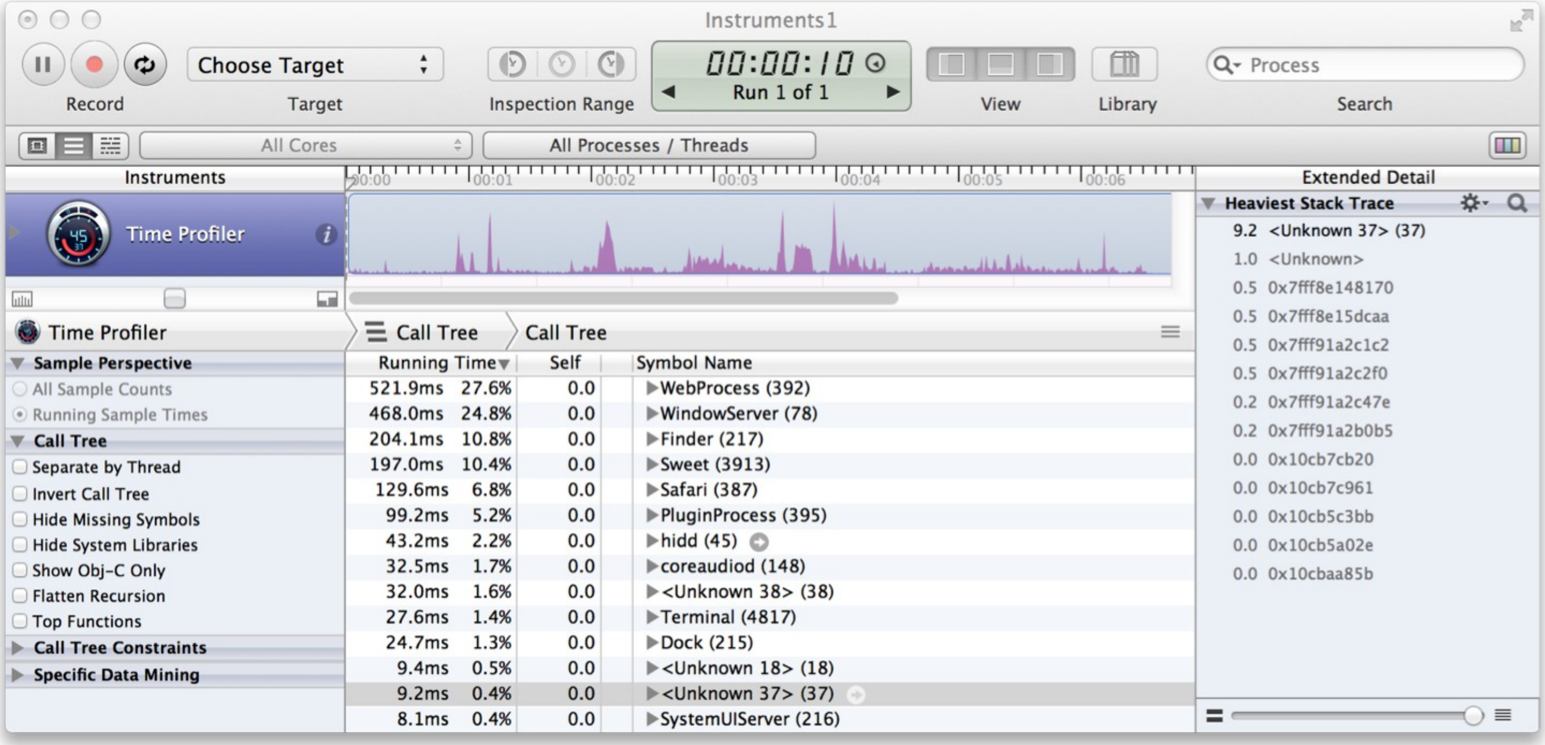
Simon Völcker

Agenda

- About Instruments
- Features
  - Templates for common tasks
  - Customization for not-so-common tasks
- Demo
  - Usage of *Grand Central Dispatch*
- Limitations
- Conclusion

Instruments

- A profiling tool available for MacOS X



Time Profiler

Sample Perspective

○ All Sample Counts
● Running Sample Times

Call Tree

□ Separate by Thread
□ Invert Call Tree
□ Hide Missing Symbols
□ Hide System Libraries
□ Show Obj-C Only
□ Flatten Recursion
□ Top Functions

▶ **Call Tree Constraints**

▶ **Specific Data Mining**

Running Time		Self	Symbol Name
521.9ms	27.6%	0.0	▶WebProcess (392)
468.0ms	24.8%	0.0	▶WindowServer (78)
204.1ms	10.8%	0.0	▶Finder (217)
197.0ms	10.4%	0.0	▶Sweet (3913)
129.6ms	6.8%	0.0	▶Safari (387)
99.2ms	5.2%	0.0	▶PluginProcess (395)
43.2ms	2.2%	0.0	▶hidd (45)
32.5ms	1.7%	0.0	▶coreaudiod (148)
32.0ms	1.6%	0.0	▶<Unknown 38> (38)
27.6ms	1.4%	0.0	▶Terminal (4817)
24.7ms	1.3%	0.0	▶Dock (215)
9.4ms	0.5%	0.0	▶<Unknown 18> (18)
9.2ms	0.4%	0.0	▶<Unknown 37> (37)
8.1ms	0.4%	0.0	▶SystemUIServer (216)

Heaviest Stack Trace

9.2	<Unknown 37> (37)
1.0	<Unknown>
0.5	0x7fff8e148170
0.5	0x7fff8e15dcaa
0.5	0x7fff91a2c1c2
0.5	0x7fff91a2c2f0
0.2	0x7fff91a2c47e
0.2	0x7fff91a2b0b5
0.0	0x10cb7cb20
0.0	0x10cb7c961
0.0	0x10cb5c3bb
0.0	0x10cb5a02e
0.0	0x10cbaa85b

- Ships with Xcode

Instruments

- Target platforms:
  - iOS, iOS Simulator, Mac OS X
- Profiling subjects:
  - CPU activity of processes and threads
  - Memory allocation/release, GC, memory leaks
  - File reads, writes, locks
  - Network activity and traffic
  - User input events (keyboard, mouse)
  - Graphics (OpenGL)

Instruments

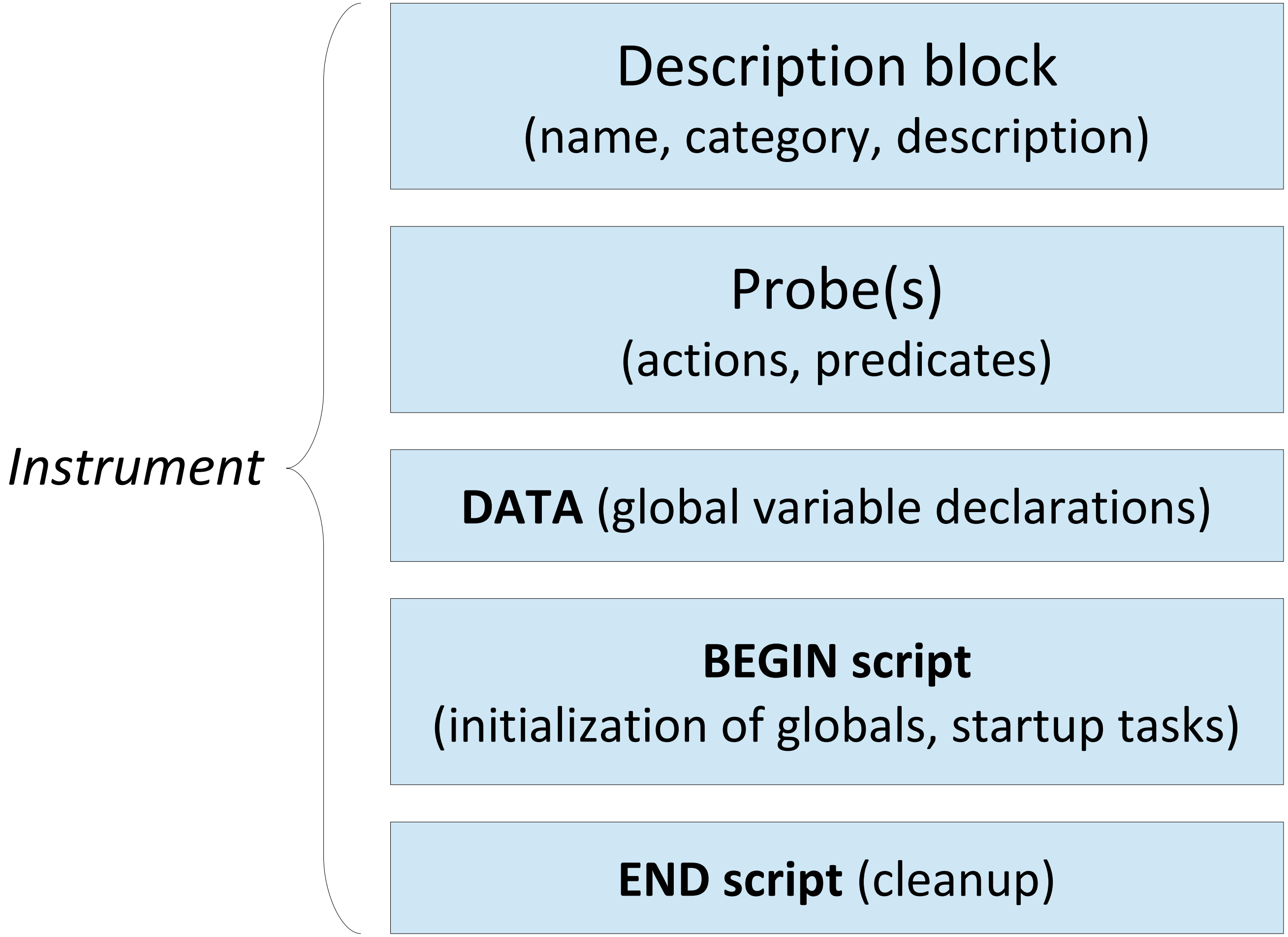
- Some data is collected via *DTrace*
  - Ported to Mac OS in v10.5 (Leopard)
- Recap on DTrace:
  - Presented 3 weeks ago
  - *Powerful* dynamic tracing framework
  - Uses *probes* – *provider:module:function:name*
  - When a probe fires, a script is run
  - Scripts usually aggregate samples and print text

Instruments

- DTrace is not available on iOS
  - Another source of data must exist
  - Sadly, no documentation on these internals

- Profiling an iOS-App from XCode provides a hint:
  - *„Build for profiling“* → Instrumentation approach

What an Instrument is



Instrument

Description block
(name, category, description)

Probe(s)
(actions, predicates)

DATA (global variable declarations)

BEGIN script
(initialization of globals, startup tasks)

END script (cleanup)

Templates

Instruments

- Rich Template library for common profiling Tasks
- Grouped by category
  - Memory
  - CPU
  - I/O Activity
  - File System
  - Graphics
  - Behavior
- Grouped by target platform
  - iOS, iOS Simulator, Mac OS X

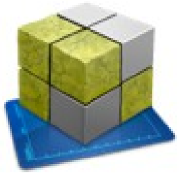
Memory usage/waste

Allocations

- Track heap memory allocations
- Object allocations

Leaks

- Detect leaked memory
- Memory address histories



Zombies

- Detect over-released objects

GC Monitor

- Analyze the object graph of a process
- Provides roots, references and allocation histories

CPU usage

Time profiler
- Sample processes in regular intervals
- Low overhead

Multicore
- Analyze multicore performance
- Threads states, dispatch queues, block usage

Dispatch
- Monitor dispatch queue activity (GCD)
- Record block invocations and their duration

I/O activity + File system

File activity

- Monitor file/directory activity
- File open/close, permission modification
- Directory creation, file moves, ...

Network

- Analyze the use of TCP/IP and UDP/IP connections
- Bytes sent/received; connection types, IPs
- Packets, roundtrip staticstics, ...

Core data

- Monitor file system activity
- Fetches, cache misses, saves

Graphics

Core animation

- Graphics performance (FPS)
- CPU usage

OpenGL ES

- Device Utilization, GL wait time
- Texture count, GL context count, …
- Count batches, enables, disables, flushes, GL calls

Behavior

UI Recorder

- Record user interface events
- Can be played back to reproduce a state

Sudden termination

- Mac OS mechanism to support faster shutdown
- Checks if sudden termination is supported by an app

Even more

Counters

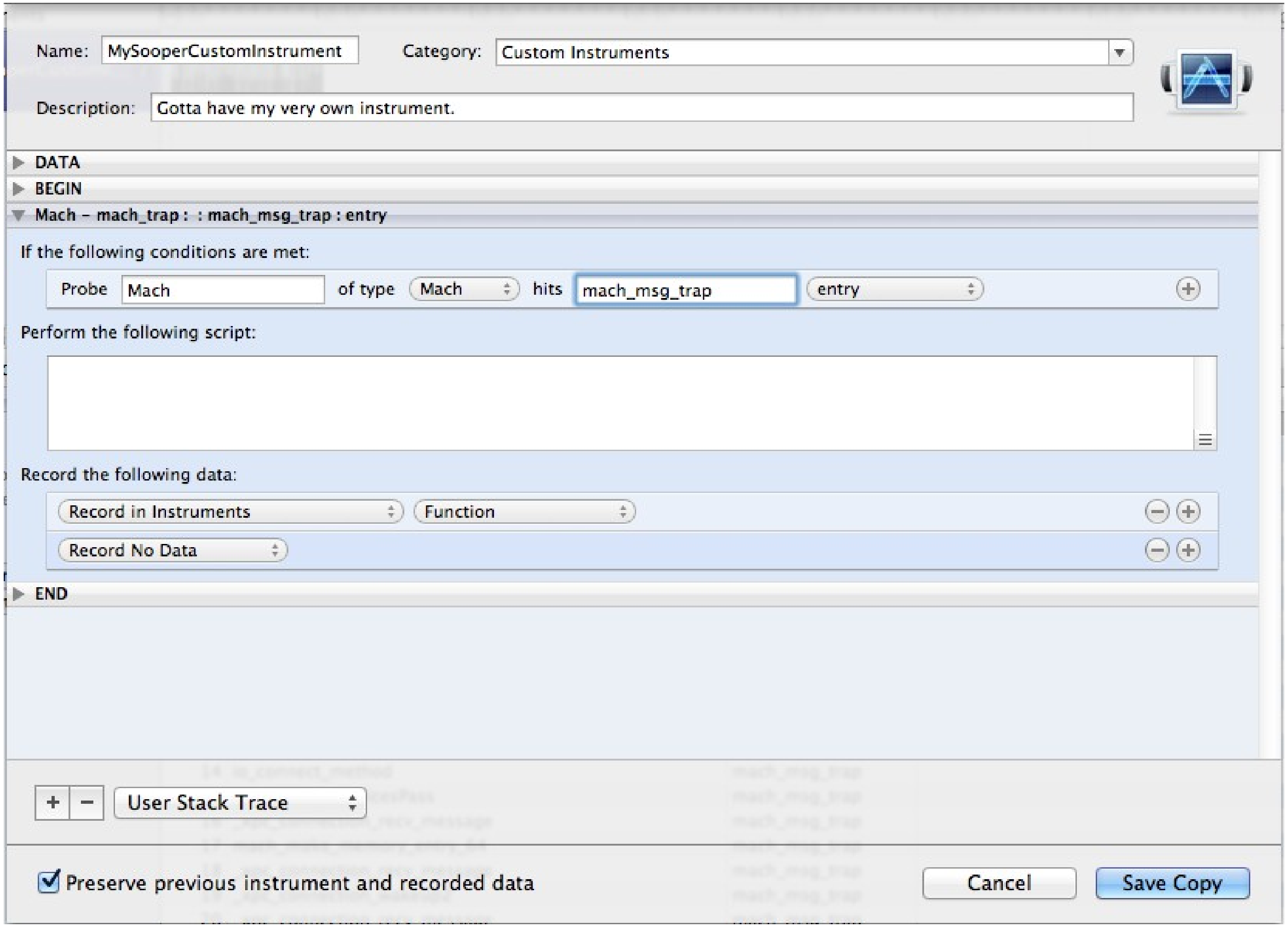
Energy diagnostics

Event profiler

System trace

Automation

Custom Instruments



Name: MySooperCustomInstrument

Category: Custom Instruments

Description: Gotta have my very own instrument.

▶ DATA

▶ BEGIN

▼ Mach – mach_trap : : mach_msg_trap : entry

If the following conditions are met:

Probe Mach of type Mach hits mach_msg_trap entry

Perform the following script:

Record the following data:

Record in Instruments | Function

Record No Data

▶ END

☑ Preserve previous instrument and recorded data

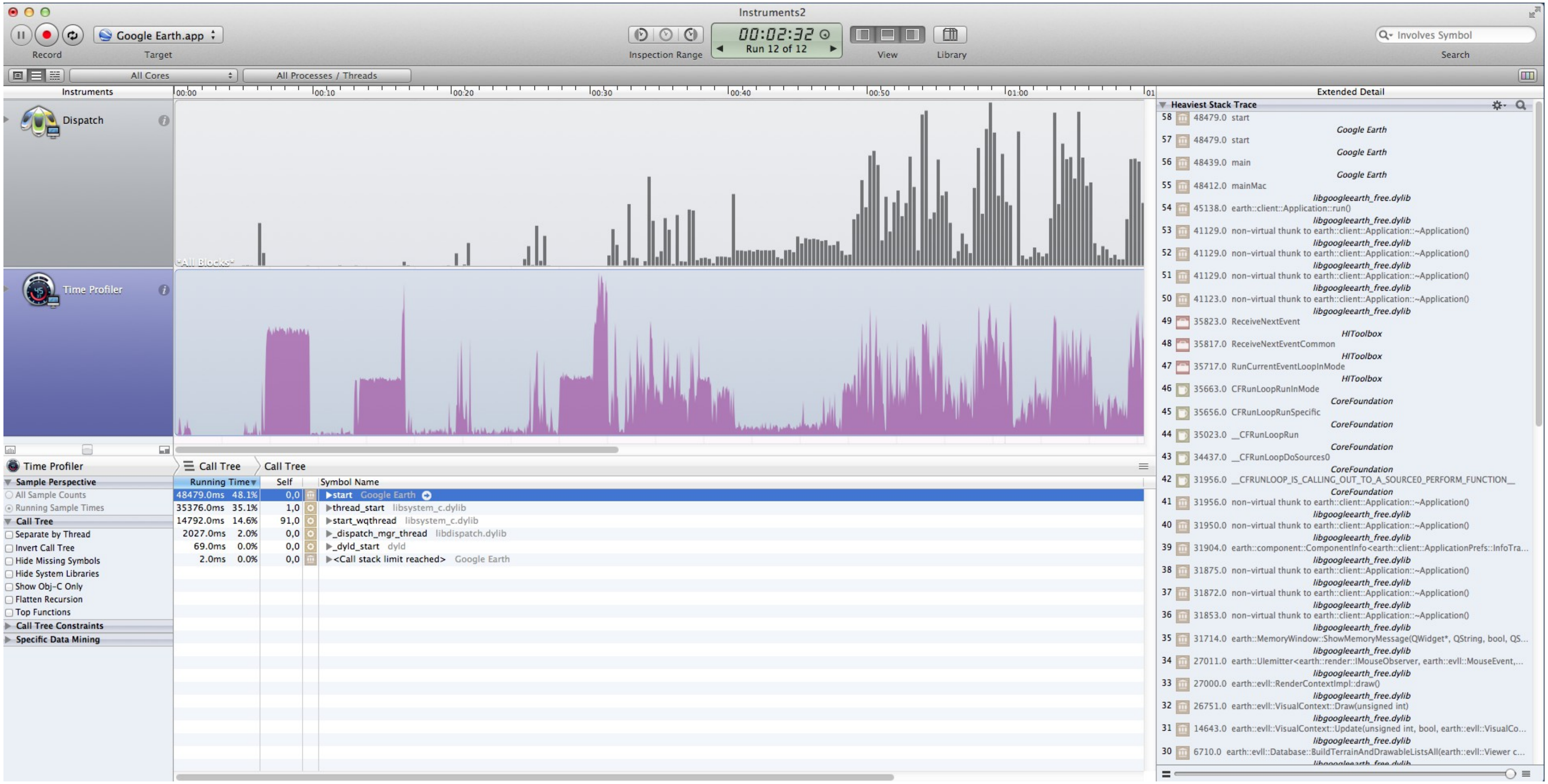
Custom Instruments

- Essentially a GUI, wrapping DTrace-scripts
  - Allows to make use of the full flexibility of DTrace
  - Is handy if an application defines its own DTrace provider
- Many providers available on Mac OS X
  - Including syscall, dtrace, mach_trap, fbt, sched
  - Use dtrace CLI to find out: /usr/sbin/dtrace
- Not possible on iOS (DTrace missing)

Demo

(i.e., screenshots...)

Demo: Profiling Google Earth



Time Profiler

▼ Sample Perspective

○ All Sample Counts

● Running Sample Times

▼ Call Tree

□ Separate by Thread

□ Invert Call Tree

□ Hide Missing Symbols

□ Hide System Libraries

□ Show Obj-C Only

□ Flatten Recursion

□ Top Functions

▶ Call Tree Constraints

▶ Specific Data Mining

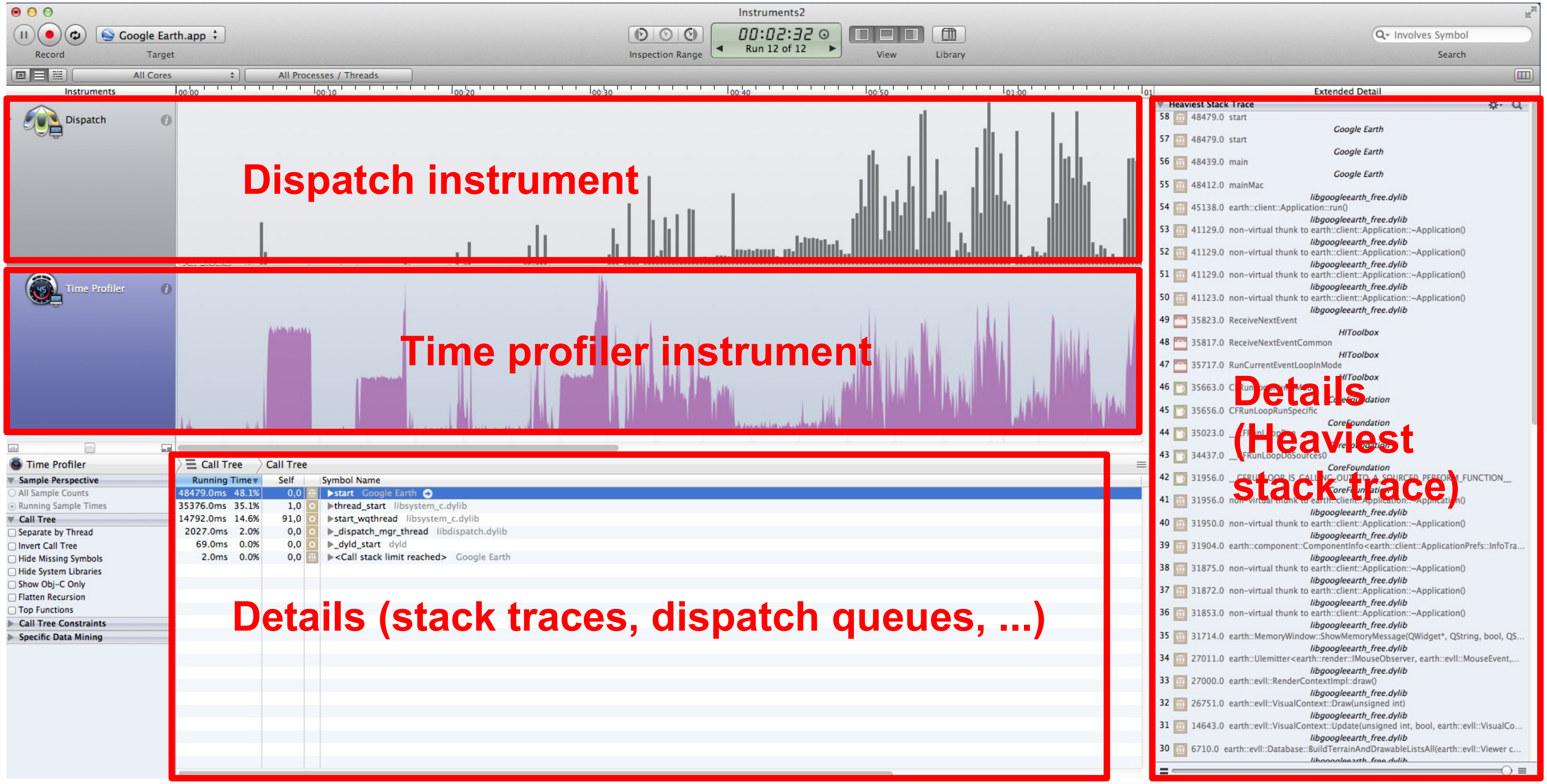
Call Tree > Call Tree

Running Time▼		Self		Symbol Name
48479.0ms	48.1%	0,0		▶start Google Earth
35376.0ms	35.1%	1,0		▶thread_start libsystem_c.dylib
14792.0ms	14.6%	91,0		▶start_wqthread libsystem_c.dylib
2027.0ms	2.0%	0,0		▶_dispatch_mgr_thread libdispatch.dylib
69.0ms	0.0%	0,0		▶_dyld_start dyld
2.0ms	0.0%	0,0		▶<Call stack limit reached> Google Earth

▼ Heaviest Stack Trace

```
58  48479.0 start
                    Google Earth
57  48479.0 start
                    Google Earth
56  48439.0 main
                    Google Earth
55  48412.0 mainMac
                    libgoogleearth_free.dylib
54  45138.0 earth::client::Application::run()
                    libgoogleearth_free.dylib
53  41129.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
52  41129.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
51  41129.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
50  41123.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
49  35823.0 ReceiveNextEvent
                    HIToolbox
48  35817.0 ReceiveNextEventCommon
                    HIToolbox
47  35717.0 RunCurrentEventLoopInMode
                    HIToolbox
46  35663.0 CFRunLoopRunInMode
                    CoreFoundation
45  35656.0 CFRunLoopRunSpecific
                    CoreFoundation
44  35023.0 __CFRunLoopRun
                    CoreFoundation
43  34437.0 __CFRunLoopDoSources0
                    CoreFoundation
42  31956.0 __CFRUNLOOP_IS_CALLING_OUT_TO_A_SOURCE0_PERFORM_FUNCTION__
                    CoreFoundation
41  31956.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
40  31950.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
39  31904.0 earth::component::ComponentInfo<earth::client::ApplicationPrefs::InfoTra...
                    libgoogleearth_free.dylib
38  31875.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
37  31872.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
36  31853.0 non-virtual thunk to earth::client::Application::~Application()
                    libgoogleearth_free.dylib
35  31714.0 earth::MemoryWindow::ShowMemoryMessage(QWidget*, QString, bool, QS...
                    libgoogleearth_free.dylib
34  27011.0 earth::UIemitter<earth::render::IMouseObserver, earth::evll::MouseEvent,...
                    libgoogleearth_free.dylib
33  27000.0 earth::evll::RenderContextImpl::draw()
                    libgoogleearth_free.dylib
32  26751.0 earth::evll::VisualContext::Draw(unsigned int)
                    libgoogleearth_free.dylib
31  14643.0 earth::evll::VisualContext::Update(unsigned int, bool, earth::evll::VisualCo...
                    libgoogleearth_free.dylib
30  6710.0 earth::evll::Database::BuildTerrainAndDrawableListsAll(earth::evll::Viewer c...
                    libgoogleearth_free.dylib
```

Instruments



Dispatch instrument

Time profiler instrument

Details (stack traces, dispatch queues, ...)

Details (Heaviest stack trace)

Time Profiler

▼ Sample Perspective
- ○ All Sample Counts
- ● Running Sample Times

▼ Call Tree
- □ Separate by Thread
- □ Invert Call Tree
- □ Hide Missing Symbols
- □ Hide System Libraries
- □ Show Obj-C Only
- □ Flatten Recursion
- □ Top Functions

▶ Call Tree Constraints

▶ Specific Data Mining

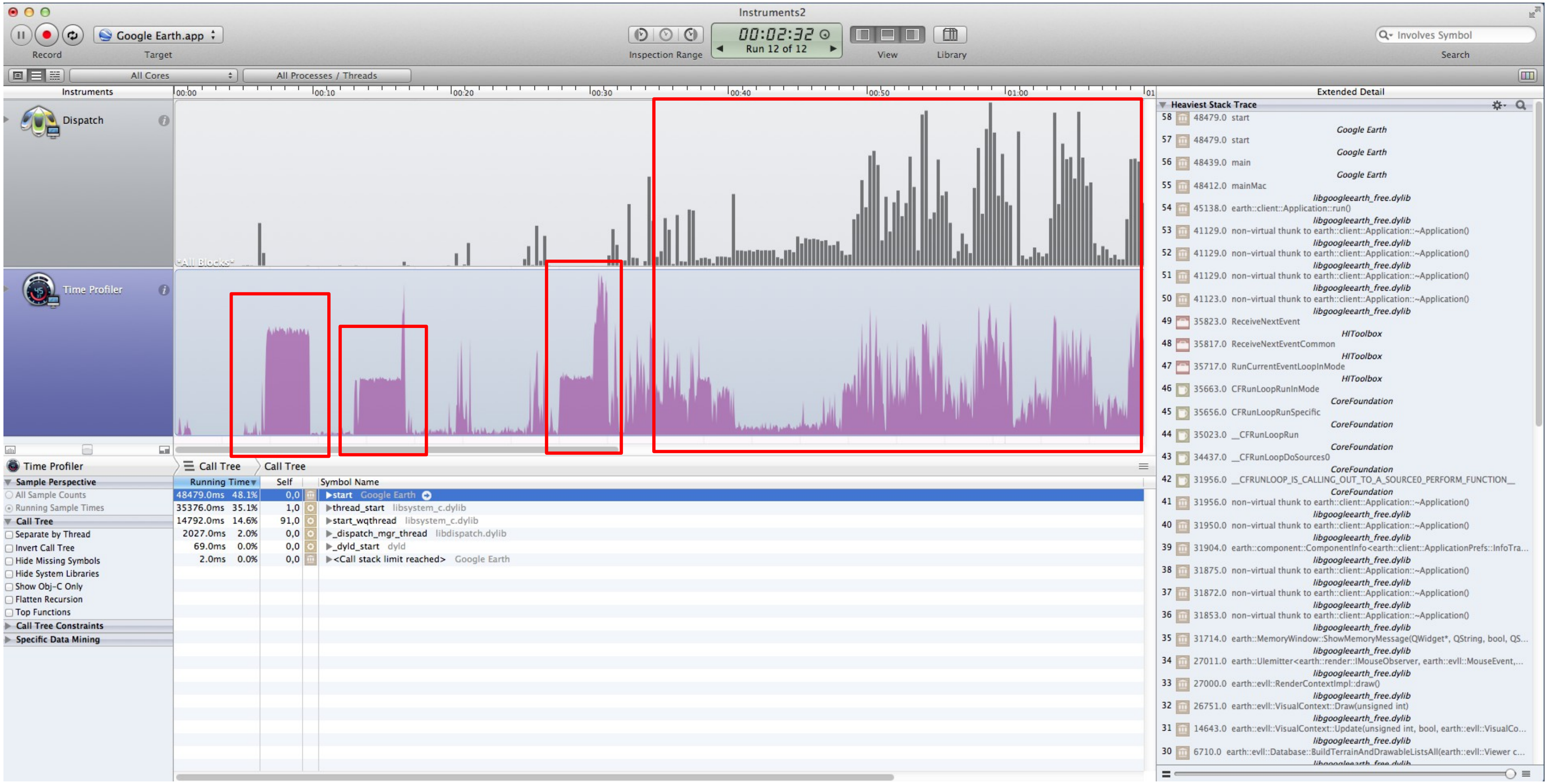
Call Tree 〉 Call Tree

Running Time ▾		Self	Symbol Name
48479.0ms	48.1%	0,0	▶start Google Earth
35376.0ms	35.1%	1,0	▶thread_start libsystem_c.dylib
14792.0ms	14.6%	91,0	▶start_wqthread libsystem_c.dylib
2027.0ms	2.0%	0,0	▶_dispatch_mgr_thread libdispatch.dylib
69.0ms	0.0%	0,0	▶_dyld_start dyld
2.0ms	0.0%	0,0	▶<Call stack limit reached> Google Earth

▼ Heaviest Stack Trace

```
58  48479.0  start
                         Google Earth
57  48479.0  start
                         Google Earth
56  48439.0  main
                         Google Earth
55  48412.0  mainMac
                         libgoogleearth_free.dylib
54  45138.0  earth::client::Application::run()
                         libgoogleearth_free.dylib
53  41129.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
52  41129.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
51  41129.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
50  41123.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
49  35823.0  ReceiveNextEvent
                         HIToolbox
48  35817.0  ReceiveNextEventCommon
                         HIToolbox
47  35717.0  RunCurrentEventLoopInMode
                         HIToolbox
46  35663.0  CFRunLoopRunInMode
                         CoreFoundation
45  35656.0  CFRunLoopRunSpecific
                         CoreFoundation
44  35023.0  __CFRunLoopRun
                         CoreFoundation
43  34437.0  __CFRunLoopDoSources0
                         CoreFoundation
42  31956.0  __CFRUNLOOP_IS_CALLING_OUT_TO_A_SOURCE0_PERFORM_FUNCTION__
                         CoreFoundation
41  31956.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
40  31950.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
39  31904.0  earth::component::ComponentInfo<earth::client::ApplicationPrefs::InfoTra...
                         libgoogleearth_free.dylib
38  31875.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
37  31872.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
36  31853.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
35  31714.0  earth::MemoryWindow::ShowMemoryMessage(QWidget*, QString, bool, QS...
                         libgoogleearth_free.dylib
34  27011.0  earth::UIemitter<earth::render::IMouseObserver, earth::evll::MouseEvent,...
                         libgoogleearth_free.dylib
33  27000.0  earth::evll::RenderContextImpl::draw()
                         libgoogleearth_free.dylib
32  26751.0  earth::evll::VisualContext::Draw(unsigned int)
                         libgoogleearth_free.dylib
31  14643.0  earth::evll::VisualContext::Update(unsigned int, bool, earth::evll::VisualCo...
                         libgoogleearth_free.dylib
30  6710.0   earth::evll::Database::BuildTerrainAndDrawableListsAll(earth::evll::Viewer c...
                         libgoogleearth_free.dylib
```

Profiling Google Earth



Time Profiler

Sample Perspective
- ○ All Sample Counts
- ● Running Sample Times

Call Tree
- □ Separate by Thread
- □ Invert Call Tree
- □ Hide Missing Symbols
- □ Hide System Libraries
- □ Show Obj-C Only
- □ Flatten Recursion
- □ Top Functions

Call Tree Constraints

Specific Data Mining

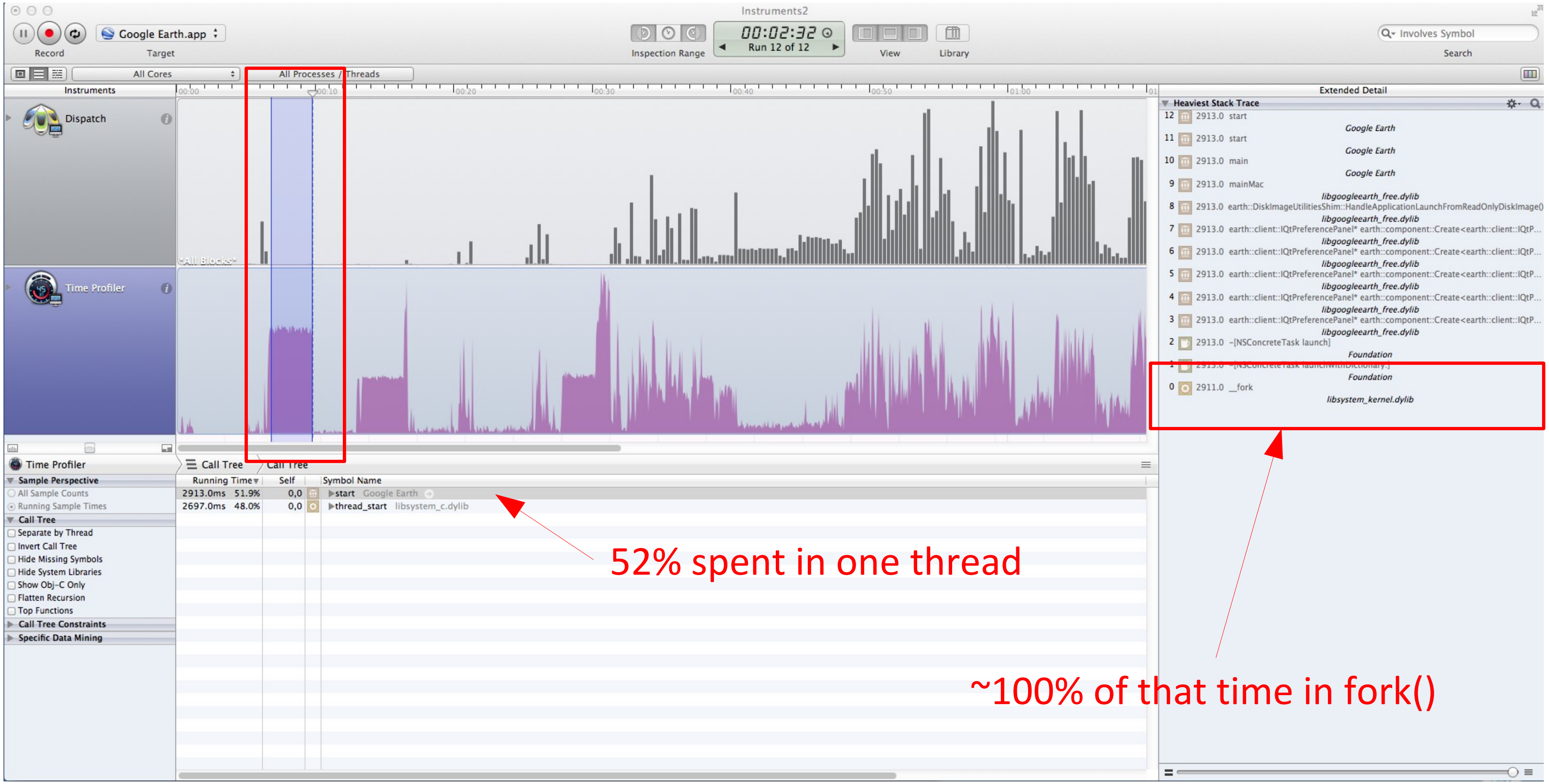
Call Tree > Call Tree

Running Time▼		Self	Symbol Name
48479.0ms	48.1%	0,0	▶start Google Earth
35376.0ms	35.1%	1,0	▶thread_start libsystem_c.dylib
14792.0ms	14.6%	91,0	▶start_wqthread libsystem_c.dylib
2027.0ms	2.0%	0,0	▶_dispatch_mgr_thread libdispatch.dylib
69.0ms	0.0%	0,0	▶_dyld_start dyld
2.0ms	0.0%	0,0	▶<Call stack limit reached> Google Earth

Heaviest Stack Trace

```
58  48479.0  start
                                   Google Earth
57  48479.0  start
                                   Google Earth
56  48439.0  main
                                   Google Earth
55  48412.0  mainMac
                                   libgoogleearth_free.dylib
54  45138.0  earth::client::Application::run()
                                   libgoogleearth_free.dylib
53  41129.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
52  41129.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
51  41129.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
50  41123.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
49  35823.0  ReceiveNextEvent
                                   HIToolbox
48  35817.0  ReceiveNextEventCommon
                                   HIToolbox
47  35717.0  RunCurrentEventLoopInMode
                                   HIToolbox
46  35663.0  CFRunLoopRunInMode
                                   CoreFoundation
45  35656.0  CFRunLoopRunSpecific
                                   CoreFoundation
44  35023.0  __CFRunLoopRun
                                   CoreFoundation
43  34437.0  __CFRunLoopDoSources0
                                   CoreFoundation
42  31956.0  __CFRUNLOOP_IS_CALLING_OUT_TO_A_SOURCE0_PERFORM_FUNCTION__
                                   CoreFoundation
41  31956.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
40  31950.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
39  31904.0  earth::component::ComponentInfo<earth::client::ApplicationPrefs::InfoTra...
                                   libgoogleearth_free.dylib
38  31875.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
37  31872.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
36  31853.0  non-virtual thunk to earth::client::Application::~Application()
                                   libgoogleearth_free.dylib
35  31714.0  earth::MemoryWindow::ShowMemoryMessage(QWidget*, QString, bool, QS...
                                   libgoogleearth_free.dylib
34  27011.0  earth::UIemitter<earth::render::IMouseObserver, earth::evll::MouseEvent,...
                                   libgoogleearth_free.dylib
33  27000.0  earth::evll::RenderContextImpl::draw()
                                   libgoogleearth_free.dylib
32  26751.0  earth::evll::VisualContext::Draw(unsigned int)
                                   libgoogleearth_free.dylib
31  14643.0  earth::evll::VisualContext::Update(unsigned int, bool, earth::evll::VisualCo...
                                   libgoogleearth_free.dylib
30  6710.0  earth::evll::Database::BuildTerrainAndDrawableListsAll(earth::evll::Viewer c...
                                   libgoogleearth_free.dylib
```

Profiling Google Earth



Time Profiler

▼ Sample Perspective
○ All Sample Counts
● Running Sample Times

▼ Call Tree
□ Separate by Thread
□ Invert Call Tree
□ Hide Missing Symbols
□ Hide System Libraries
□ Show Obj-C Only
□ Flatten Recursion
□ Top Functions

▶ Call Tree Constraints
▶ Specific Data Mining

Call Tree › Call Tree

Running Time ▾		Self	Symbol Name
2913.0ms	51.9%	0,0	▶start Google Earth
2697.0ms	48.0%	0,0	▶thread_start libsystem_c.dylib

Extended Detail

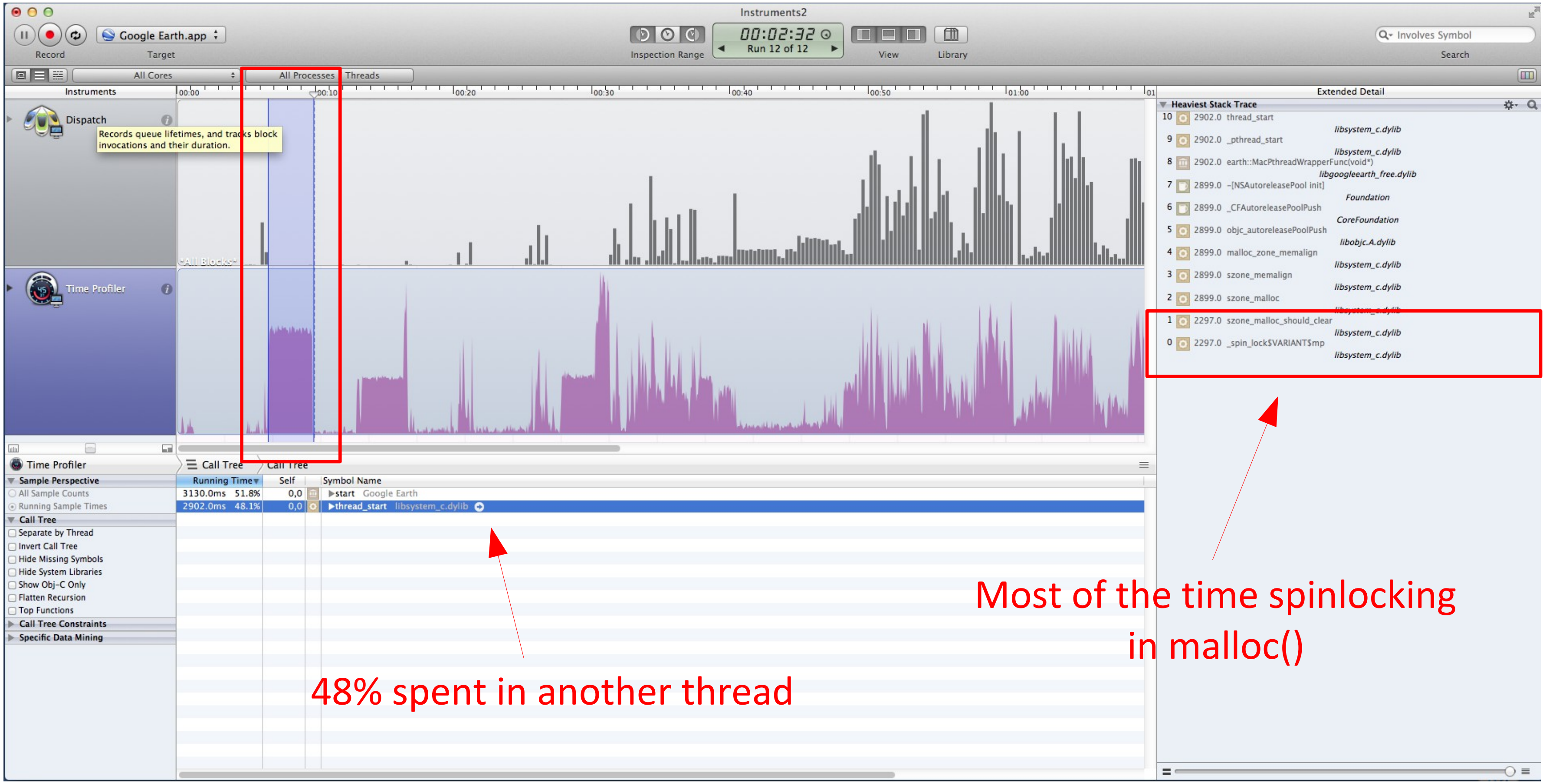
▼ Heaviest Stack Trace

```
12  2913.0  start
                Google Earth
11  2913.0  start
                Google Earth
10  2913.0  main
                Google Earth
 9  2913.0  mainMac
                libgoogleearth_free.dylib
 8  2913.0  earth::DiskImageUtilitiesShim::HandleApplicationLaunchFromReadOnlyDiskImage()
                libgoogleearth_free.dylib
 7  2913.0  earth::client::IQtPreferencePanel* earth::component::Create<earth::client::IQtP...
                libgoogleearth_free.dylib
 6  2913.0  earth::client::IQtPreferencePanel* earth::component::Create<earth::client::IQtP...
                libgoogleearth_free.dylib
 5  2913.0  earth::client::IQtPreferencePanel* earth::component::Create<earth::client::IQtP...
                libgoogleearth_free.dylib
 4  2913.0  earth::client::IQtPreferencePanel* earth::component::Create<earth::client::IQtP...
                libgoogleearth_free.dylib
 3  2913.0  earth::client::IQtPreferencePanel* earth::component::Create<earth::client::IQtP...
                libgoogleearth_free.dylib
 2  2913.0  -[NSConcreteTask launch]
                Foundation
 1  2913.0  -[NSConcreteTask launchWithDictionary:]
                Foundation
 0  2911.0  __fork
                libsystem_kernel.dylib
```

52% spent in one thread

~100% of that time in fork()

Profiling Google Earth



Time Profiler

▼ Sample Perspective

○ All Sample Counts

● Running Sample Times

▼ Call Tree

□ Separate by Thread

□ Invert Call Tree

□ Hide Missing Symbols

□ Hide System Libraries

□ Show Obj-C Only

□ Flatten Recursion

□ Top Functions

▶ Call Tree Constraints

▶ Specific Data Mining

Call Tree › Call Tree

Running Time▼		Self	Symbol Name
3130.0ms	51.8%	0,0	▶start Google Earth
2902.0ms	48.1%	0,0	▶thread_start libsystem_c.dylib

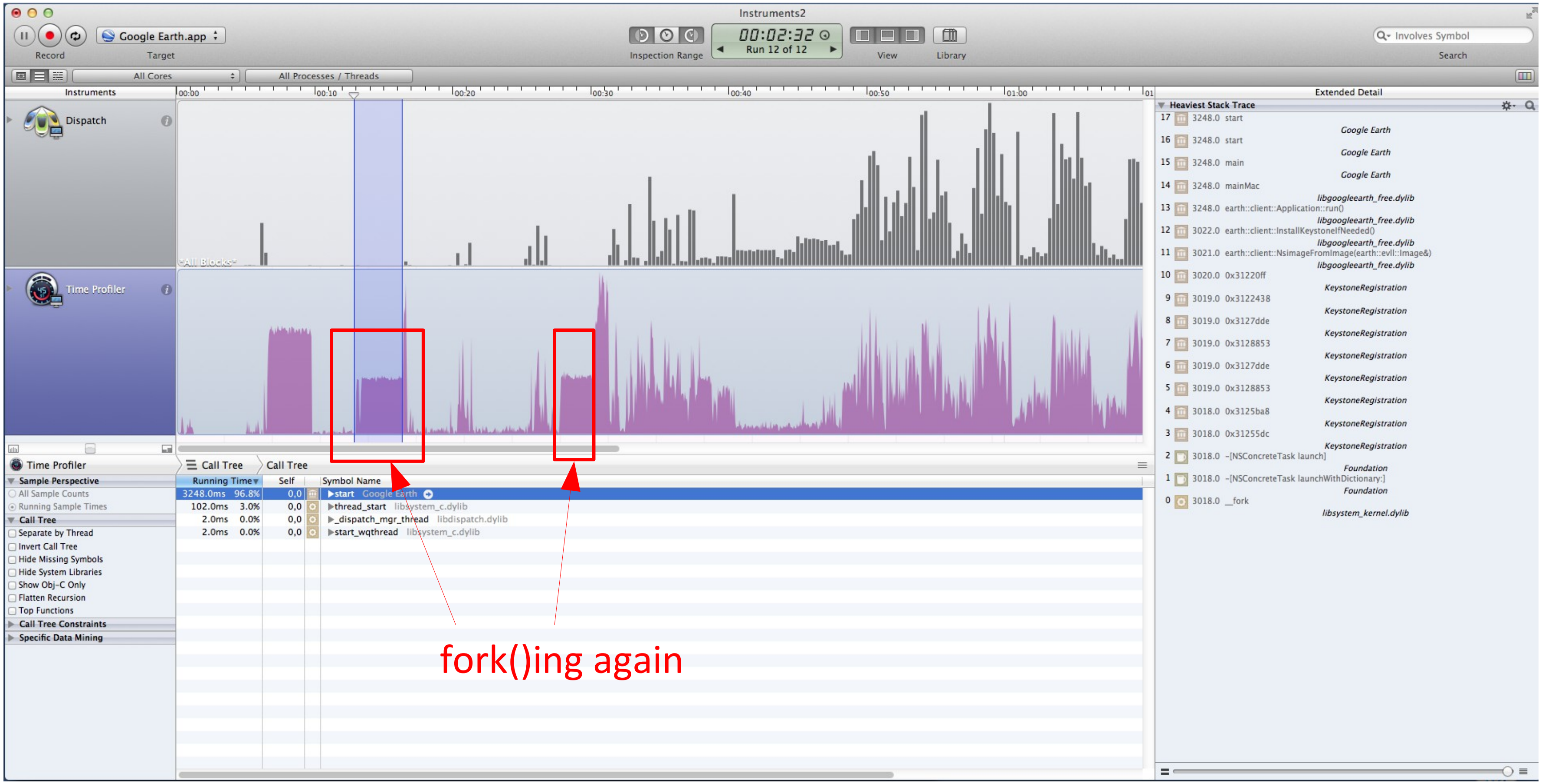
▼ Heaviest Stack Trace

10	2902.0	thread_start	libsystem_c.dylib
9	2902.0	_pthread_start	libsystem_c.dylib
8	2902.0	earth::MacPthreadWrapperFunc(void*)	libgoogleearth_free.dylib
7	2899.0	-[NSAutoreleasePool init]	Foundation
6	2899.0	_CFAutoreleasePoolPush	CoreFoundation
5	2899.0	objc_autoreleasePoolPush	libobjc.A.dylib
4	2899.0	malloc_zone_memalign	libsystem_c.dylib
3	2899.0	szone_memalign	libsystem_c.dylib
2	2899.0	szone_malloc	libsystem_c.dylib
1	2297.0	szone_malloc_should_clear	libsystem_c.dylib
0	2297.0	_spin_lock$VARIANT$mp	libsystem_c.dylib

Most of the time spinlocking in malloc()

48% spent in another thread

Profiling Google Earth



Time Profiler

▼ Sample Perspective
○ All Sample Counts
● Running Sample Times
▼ Call Tree
□ Separate by Thread
□ Invert Call Tree
□ Hide Missing Symbols
□ Hide System Libraries
□ Show Obj-C Only
□ Flatten Recursion
□ Top Functions
▶ Call Tree Constraints
▶ Specific Data Mining

Call Tree > Call Tree

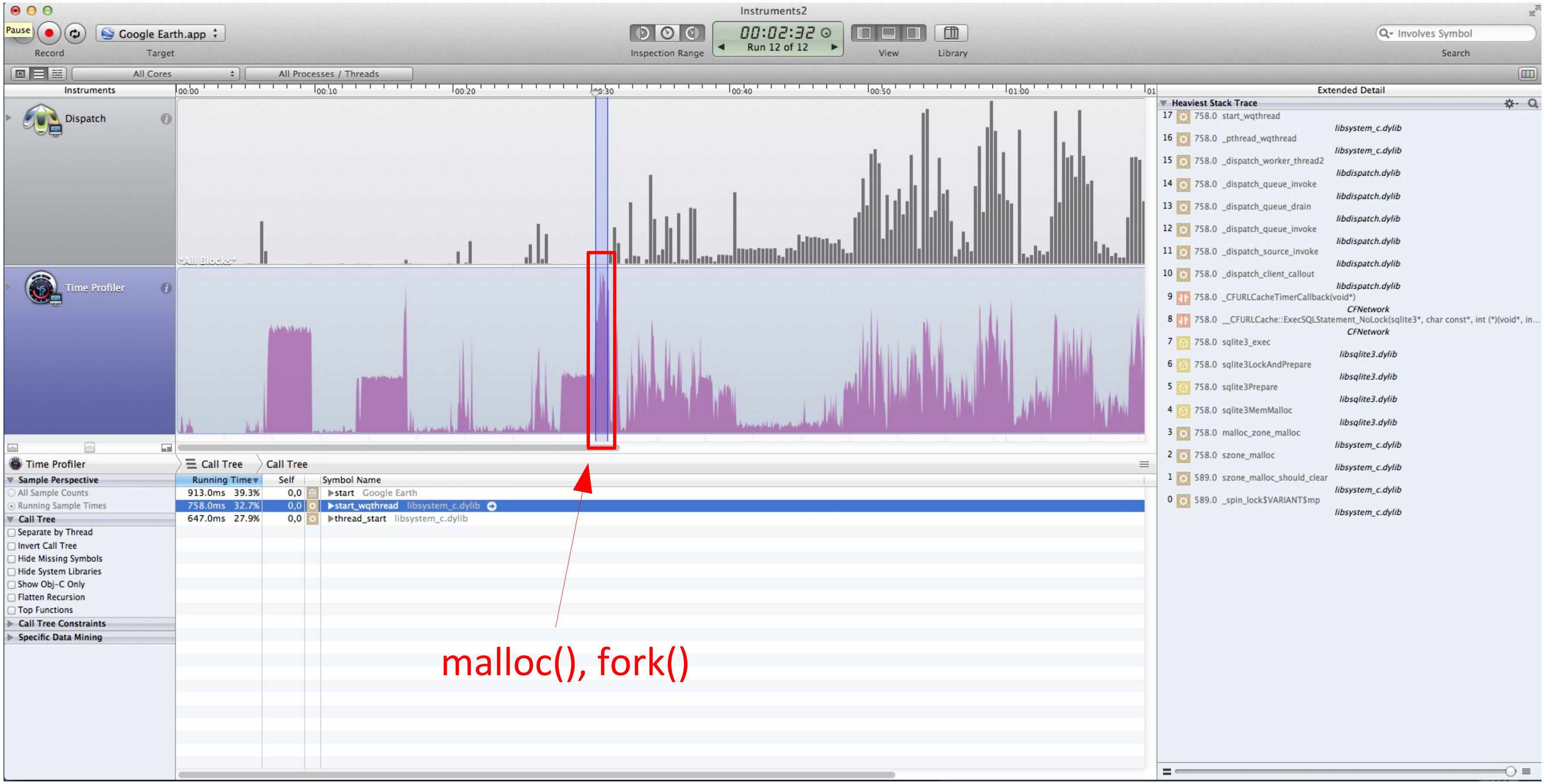
Running Time		Self	Symbol Name
3248.0ms	96.8%	0,0	▶start Google Earth
102.0ms	3.0%	0,0	▶thread_start libsystem_c.dylib
2.0ms	0.0%	0,0	▶_dispatch_mgr_thread libdispatch.dylib
2.0ms	0.0%	0,0	▶start_wqthread libsystem_c.dylib

▼ Heaviest Stack Trace

```
17  3248.0  start
                          Google Earth
16  3248.0  start
                          Google Earth
15  3248.0  main
                          Google Earth
14  3248.0  mainMac
                          libgoogleearth_free.dylib
13  3248.0  earth::client::Application::run()
                          libgoogleearth_free.dylib
12  3022.0  earth::client::InstallKeystoneIfNeeded()
                          libgoogleearth_free.dylib
11  3021.0  earth::client::NsimageFromImage(earth::evll::Image&)
                          libgoogleearth_free.dylib
10  3020.0  0x31220ff
                          KeystoneRegistration
 9  3019.0  0x3122438
                          KeystoneRegistration
 8  3019.0  0x3127dde
                          KeystoneRegistration
 7  3019.0  0x3128853
                          KeystoneRegistration
 6  3019.0  0x3127dde
                          KeystoneRegistration
 5  3019.0  0x3128853
                          KeystoneRegistration
 4  3018.0  0x3125ba8
                          KeystoneRegistration
 3  3018.0  0x31255dc
                          KeystoneRegistration
 2  3018.0  -[NSConcreteTask launch]
                          Foundation
 1  3018.0  -[NSConcreteTask launchWithDictionary:]
                          Foundation
 0  3018.0  __fork
                          libsystem_kernel.dylib
```

fork()ing again

Profiling Google Earth



Time Profiler

▼ Sample Perspective
○ All Sample Counts
● Running Sample Times

▼ Call Tree
□ Separate by Thread
□ Invert Call Tree
□ Hide Missing Symbols
□ Hide System Libraries
□ Show Obj-C Only
□ Flatten Recursion
□ Top Functions

▶ Call Tree Constraints
▶ Specific Data Mining

Call Tree > Call Tree

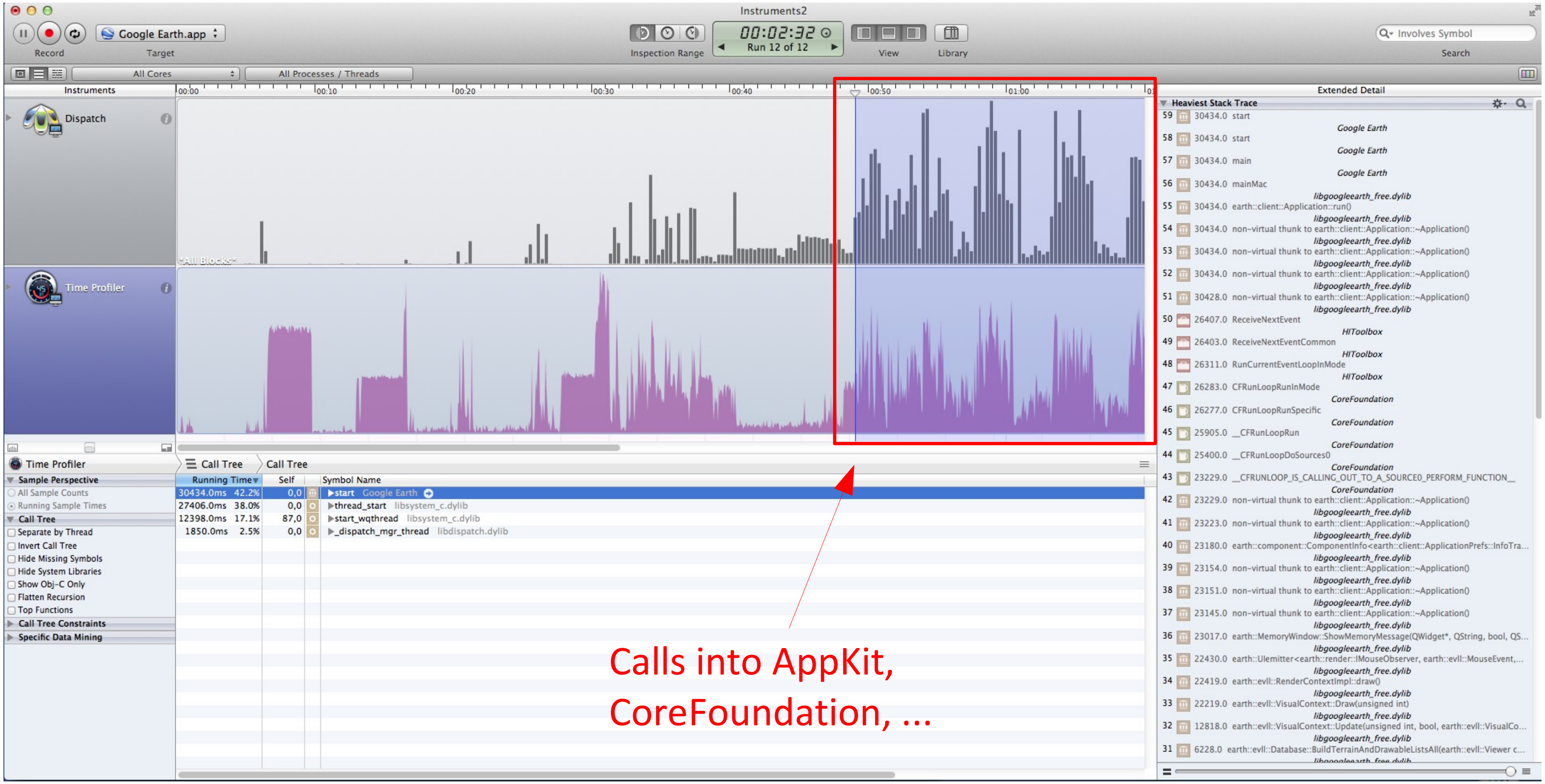
Running Time		Self	Symbol Name
913.0ms	39.3%	0,0	▶start Google Earth
758.0ms	32.7%	0,0	▶start_wqthread libsystem_c.dylib
647.0ms	27.9%	0,0	▶thread_start libsystem_c.dylib

Heaviest Stack Trace

```
17  758.0  start_wqthread
                                   libsystem_c.dylib
16  758.0  _pthread_wqthread
                                   libsystem_c.dylib
15  758.0  _dispatch_worker_thread2
                                   libdispatch.dylib
14  758.0  _dispatch_queue_invoke
                                   libdispatch.dylib
13  758.0  _dispatch_queue_drain
                                   libdispatch.dylib
12  758.0  _dispatch_queue_invoke
                                   libdispatch.dylib
11  758.0  _dispatch_source_invoke
                                   libdispatch.dylib
10  758.0  _dispatch_client_callout
                                   libdispatch.dylib
 9  758.0  _CFURLCacheTimerCallback(void*)
                                   CFNetwork
 8  758.0  __CFURLCache::ExecSQLStatement_NoLock(sqlite3*, char const*, int (*)(void*, in...
                                   CFNetwork
 7  758.0  sqlite3_exec
                                   libsqlite3.dylib
 6  758.0  sqlite3LockAndPrepare
                                   libsqlite3.dylib
 5  758.0  sqlite3Prepare
                                   libsqlite3.dylib
 4  758.0  sqlite3MemMalloc
                                   libsqlite3.dylib
 3  758.0  malloc_zone_malloc
                                   libsystem_c.dylib
 2  758.0  szone_malloc
                                   libsystem_c.dylib
 1  589.0  szone_malloc_should_clear
                                   libsystem_c.dylib
 0  589.0  _spin_lock$VARIANT$mp
                                   libsystem_c.dylib
```

malloc(), fork()

Profiling Google Earth



Time Profiler

Sample Perspective
- ○ All Sample Counts
- ● Running Sample Times

Call Tree
- □ Separate by Thread
- □ Invert Call Tree
- □ Hide Missing Symbols
- □ Hide System Libraries
- □ Show Obj-C Only
- □ Flatten Recursion
- □ Top Functions

Call Tree Constraints

Specific Data Mining

Call Tree > Call Tree

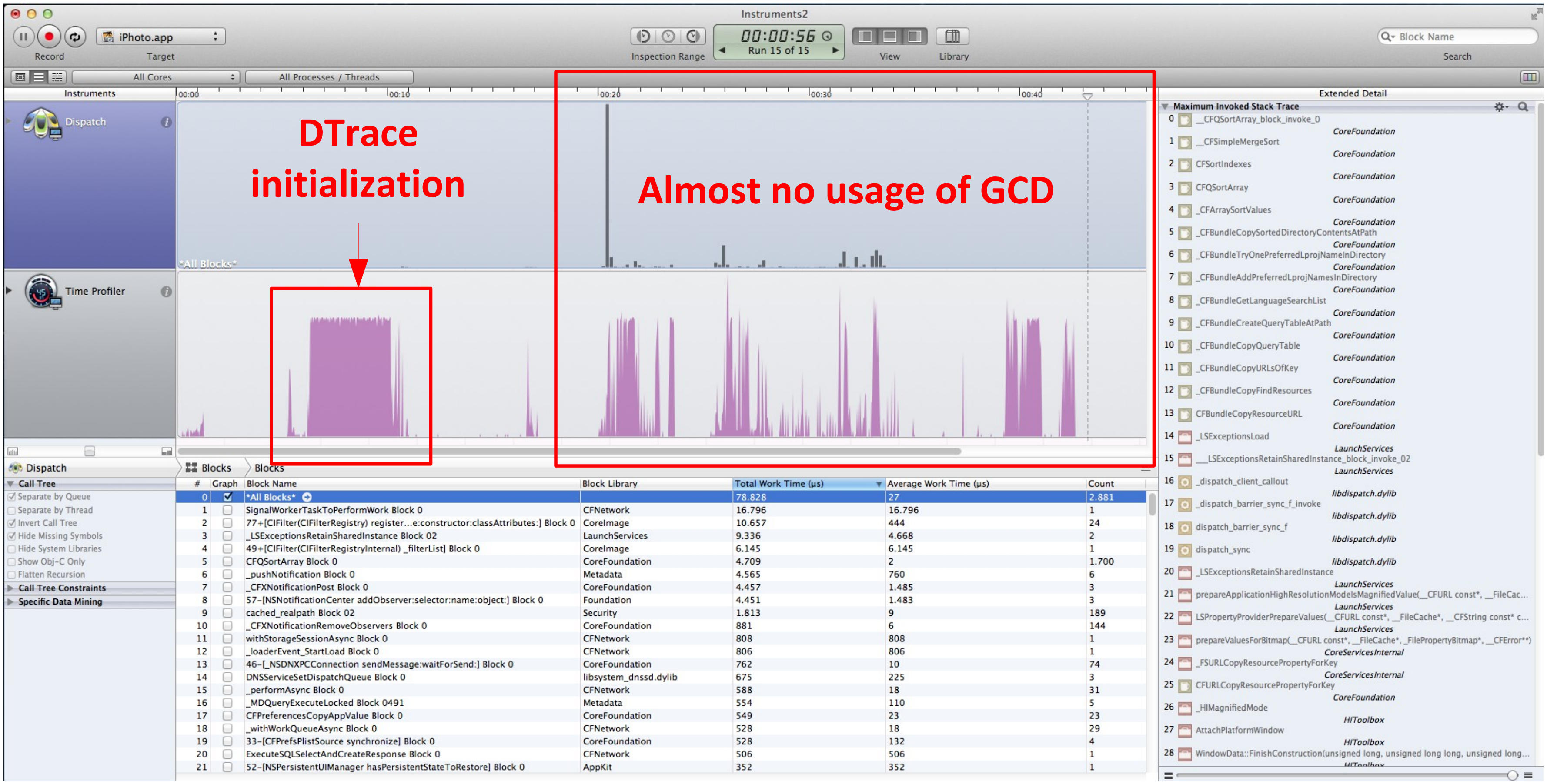
Running Time		Self	Symbol Name
30434.0ms	42.2%	0,0	▶start Google Earth
27406.0ms	38.0%	0,0	▶thread_start libsystem_c.dylib
12398.0ms	17.1%	87,0	▶start_wqthread libsystem_c.dylib
1850.0ms	2.5%	0,0	▶_dispatch_mgr_thread libdispatch.dylib

Heaviest Stack Trace

```
59  30434.0  start
                         Google Earth
58  30434.0  start
                         Google Earth
57  30434.0  main
                         Google Earth
56  30434.0  mainMac
                         libgoogleearth_free.dylib
55  30434.0  earth::client::Application::run()
                         libgoogleearth_free.dylib
54  30434.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
53  30434.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
52  30434.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
51  30428.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
50  26407.0  ReceiveNextEvent
                         HIToolbox
49  26403.0  ReceiveNextEventCommon
                         HIToolbox
48  26311.0  RunCurrentEventLoopInMode
                         HIToolbox
47  26283.0  CFRunLoopRunInMode
                         CoreFoundation
46  26277.0  CFRunLoopRunSpecific
                         CoreFoundation
45  25905.0  __CFRunLoopRun
                         CoreFoundation
44  25400.0  __CFRunLoopDoSources0
                         CoreFoundation
43  23229.0  __CFRUNLOOP_IS_CALLING_OUT_TO_A_SOURCE0_PERFORM_FUNCTION__
                         CoreFoundation
42  23229.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
41  23223.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
40  23180.0  earth::component::ComponentInfo<earth::client::ApplicationPrefs::InfoTra...
                         libgoogleearth_free.dylib
39  23154.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
38  23151.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
37  23145.0  non-virtual thunk to earth::client::Application::~Application()
                         libgoogleearth_free.dylib
36  23017.0  earth::MemoryWindow::ShowMemoryMessage(QWidget*, QString, bool, QS...
                         libgoogleearth_free.dylib
35  22430.0  earth::Ulemitter<earth::render::IMouseObserver, earth::evll::MouseEvent,...
                         libgoogleearth_free.dylib
34  22419.0  earth::evll::RenderContextImpl::draw()
                         libgoogleearth_free.dylib
33  22219.0  earth::evll::VisualContext::Draw(unsigned int)
                         libgoogleearth_free.dylib
32  12818.0  earth::evll::VisualContext::Update(unsigned int, bool, earth::evll::VisualCo...
                         libgoogleearth_free.dylib
31  6228.0   earth::evll::Database::BuildTerrainAndDrawableListsAll(earth::evll::Viewer c...
```

Calls into AppKit, CoreFoundation, ...

Profiling iPhoto



DTrace initialization

Almost no usage of GCD

Dispatch > Blocks > Blocks

Call Tree

- ☑ Separate by Queue
- ☐ Separate by Thread
- ☑ Invert Call Tree
- ☑ Hide Missing Symbols
- ☐ Hide System Libraries
- ☐ Show Obj-C Only
- ☐ Flatten Recursion

Call Tree Constraints

Specific Data Mining

#	Graph	Block Name	Block Library	Total Work Time (µs)	Average Work Time (µs)	Count
0	☑	*All Blocks*		78.828	27	2.881
1	☐	SignalWorkerTaskToPerformWork Block 0	CFNetwork	16.796	16.796	1
2	☐	77+[CIFilter(CIFilterRegistry) register...e:constructor:classAttributes:] Block 0	CoreImage	10.657	444	24
3	☐	_LSExceptionsRetainSharedInstance Block 02	LaunchServices	9.336	4.668	2
4	☐	49+[CIFilter(CIFilterRegistryInternal) _filterList] Block 0	CoreImage	6.145	6.145	1
5	☐	CFQSortArray Block 0	CoreFoundation	4.709	2	1.700
6	☐	_pushNotification Block 0	Metadata	4.565	760	6
7	☐	_CFXNotificationPost Block 0	CoreFoundation	4.457	1.485	3
8	☐	57-[NSNotificationCenter addObserver:selector:name:object:] Block 0	Foundation	4.451	1.483	3
9	☐	cached_realpath Block 02	Security	1.813	9	189
10	☐	_CFXNotificationRemoveObservers Block 0	CoreFoundation	881	6	144
11	☐	withStorageSessionAsync Block 0	CFNetwork	808	808	1
12	☐	_loaderEvent_StartLoad Block 0	CFNetwork	806	806	1
13	☐	46-[_NSDNXPCConnection sendMessage:waitForSend:] Block 0	CoreFoundation	762	10	74
14	☐	DNSServiceSetDispatchQueue Block 0	libsystem_dnssd.dylib	675	225	3
15	☐	_performAsync Block 0	CFNetwork	588	18	31
16	☐	_MDQueryExecuteLocked Block 0491	Metadata	554	110	5
17	☐	CFPreferencesCopyAppValue Block 0	CoreFoundation	549	23	23
18	☐	_withWorkQueueAsync Block 0	CFNetwork	528	18	29
19	☐	33-[CFPrefsPlistSource synchronize] Block 0	CoreFoundation	528	132	4
20	☐	ExecuteSQLSelectAndCreateResponse Block 0	CFNetwork	506	506	1
21	☐	52-[NSPersistentUIManager hasPersistentStateToRestore] Block 0	AppKit	352	352	1

Maximum Invoked Stack Trace

0	__CFQSortArray_block_invoke_0	*CoreFoundation*
1	__CFSimpleMergeSort	*CoreFoundation*
2	CFSortIndexes	*CoreFoundation*
3	CFQSortArray	*CoreFoundation*
4	_CFArraySortValues	*CoreFoundation*
5	_CFBundleCopySortedDirectoryContentsAtPath	*CoreFoundation*
6	_CFBundleTryOnePreferredLprojNameInDirectory	*CoreFoundation*
7	_CFBundleAddPreferredLprojNamesInDirectory	*CoreFoundation*
8	_CFBundleGetLanguageSearchList	*CoreFoundation*
9	_CFBundleCreateQueryTableAtPath	*CoreFoundation*
10	_CFBundleCopyQueryTable	*CoreFoundation*
11	_CFBundleCopyURLsOfKey	*CoreFoundation*
12	_CFBundleCopyFindResources	*CoreFoundation*
13	CFBundleCopyResourceURL	*CoreFoundation*
14	_LSExceptionsLoad	*LaunchServices*
15	___LSExceptionsRetainSharedInstance_block_invoke_02	*LaunchServices*
16	_dispatch_client_callout	*libdispatch.dylib*
17	_dispatch_barrier_sync_f_invoke	*libdispatch.dylib*
18	dispatch_barrier_sync_f	*libdispatch.dylib*
19	dispatch_sync	*libdispatch.dylib*
20	_LSExceptionsRetainSharedInstance	*LaunchServices*
21	prepareApplicationHighResolutionModeIsMagnifiedValue(__CFURL const*, __FileCac...	*LaunchServices*
22	LSPropertyProviderPrepareValues(__CFURL const*, __FileCache*, __CFString const* c...	*LaunchServices*
23	prepareValuesForBitmap(__CFURL const*, __FileCache*, _FilePropertyBitmap*, __CFError**)	*CoreServicesInternal*
24	_FSURLCopyResourcePropertyForKey	*CoreServicesInternal*
25	CFURLCopyResourcePropertyForKey	*CoreFoundation*
26	_HIMagnifiedMode	*HIToolbox*
27	AttachPlatformWindow	*HIToolbox*
28	WindowData::FinishConstruction(unsigned long, unsigned long long, unsigned long...	*HIToolbox*

Limitations

Just one, actually

Limitations

- OS X processes can explicitly deny tracing+debugging

```
#if defined(__APPLE__)
/*
* If the thread on which this probe has fired belongs to a process marked P_LNOATTACH
* then this enabling is not permitted to observe it. Move along, nothing to see here.
*/
if (ISSET(current_proc()->p_lflag, P_LNOATTACH)) {
continue;
}
#endif /* __APPLE__ */
```

„This is antithetical to the notion of systemic tracing, antithetical to the goals of DTrace, and antithetical to the spirit of open source.“

- Adam Leventhal (DTrace)

Official view

> „**Note**: Several Apple apps - namely, **iTunes**, **DVD Player**, and **Front Row** and **apps that use QuickTime - prevent the collection of data** through Dtrace (either temporarily or permanently) in order to **protect sensitive data**.
>
> Therefore, you should not run those apps when performing systemwide data collection.“
>
> *Instruments documentation*

- Presence of protected processes influences systemwide measurements
- Still, some Instruments seem to work with iTunes
  - Time profiler, Allocation tracker, UI events, Network monitor, …
  - Either: There is a data source aside DTrace
  - Or: iTunes is just missing in the results of all these Instruments

Conclusion

Conclusion

- Instruments...
  - Is a powerful tool for *many* profiling tasks
  - Has capabilities beyond bare profiling (Behavior)
  - Has a stunningly simple GUI
  - Is still flexible as DTrace

- On the other hand...
  - It is crippled by nature
  - Occasional malfunction spoils the overall experience
  - Only little documentation of the inner workings

References

- http://dtrace.org/blogs/ahl/2008/01/18/mac-os-x-and-the-missing-probes/
- http://en.wikipedia.org/wiki/DTrace [April 7, 2013]
- http://en.wikipedia.org/wiki/Instruments_(application) [April 12, 2013]
- http://developer.apple.com/library/mac/[...]/Introduction.html
- http://www.raywenderlich.com/23037/how-to-use-instruments-in-xcode
- http://www.symantec.com/business/support/index?page=content&id=HOWTO36824
- Mac OS X and iOS Internals: To the Apple's Core by Levin, Jonathan [10/2012]
- ECE459: Programming for Performance (patricklam.ca/p4p/2011/notes/pdf/L04.pdf)