Getting Started with Instruments

Tibet Rooney-Rabdau, Xcode Engineer
Ben Mitchell, Xcode Engineer
Anand Subramanian, Xcode Engineer
Responsiveness
Great user experience
Orientation
Profiling your app
Using Signposts
Orientation

Profiling your app

Using Signposts
<table>
<thead>
<tr>
<th>Product</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Run</td>
<td>🌟R</td>
</tr>
<tr>
<td>Test</td>
<td>🌟U</td>
</tr>
<tr>
<td>Profile</td>
<td>🌟I</td>
</tr>
<tr>
<td>Analyze</td>
<td>🌟B</td>
</tr>
<tr>
<td>Archive</td>
<td>🌟B</td>
</tr>
<tr>
<td>Build For</td>
<td></td>
</tr>
<tr>
<td>Perform Action</td>
<td></td>
</tr>
<tr>
<td>Build</td>
<td>🌟B</td>
</tr>
<tr>
<td>Clean Build Folder</td>
<td>🌟K</td>
</tr>
<tr>
<td>Stop</td>
<td>🌟.</td>
</tr>
<tr>
<td>Scheme</td>
<td></td>
</tr>
<tr>
<td>Destination</td>
<td></td>
</tr>
<tr>
<td>Test Plan</td>
<td></td>
</tr>
<tr>
<td>Create Bot</td>
<td></td>
</tr>
</tbody>
</table>
Recording in Windowed Mode
Demo
Profiling your app

Ben Mitchell, Xcode Engineer
Profiling Tips
Profiling Tips

Time Profiler shows how your app is spending time
Profiling Tips

Time Profiler shows how your app is spending time

Check main thread when responsiveness issues occur
Profiling Tips

Time Profiler shows how your app is spending time

Check main thread when responsiveness issues occur

Profile release builds
Profiling Tips

Time Profiler shows how your app is spending time

Check main thread when responsiveness issues occur

Profile release builds

Profile with difficult workloads and older devices
Instruments on All Platforms
Simulator Caveat
Simulator Caveat
Simulator Caveat
Simulator Caveat
Simulator Caveat
Simulator Caveat
What About Efficiency?

Main thread responsiveness isn’t the whole story

High CPU use can
• Drain the battery
• Heat up the device
• Spin up fans
Using Signposts

Anand Subramanian, Xcode Engineer
Statistical Profile Versus Measurement
Statistical Profile Versus Measurement
Statistical Profile Versus Measurement

Time Profiler

Execution pattern 1

Execution pattern 2
Statistical Profile Versus Measurement

Time Profiler

Execution pattern 1

Execution pattern 2

Execution pattern 3
How to log operations?
Use Signposts!
Features of Signposts
Features of Signposts

Simpler and more efficient than printing
Features of Signposts

Simpler and more efficient than printing

Built-in support for measuring time
Features of Signposts

Simpler and more efficient than printing

Built-in support for measuring time

Traced by Instruments
Signpost Intervals in Instruments
Signpost Intervals in Instruments

![Image of Instruments window with Time Profiler and Points of Interest](image-url)

- **Time Profiler**: CPU Usage
- **Points of Interest**: setupScene (com.apple.SolarSystem)
Demo

Using Signposts with Points of Interest
Concepts
Concepts

Statistical profiles show which code is most commonly executed
Concepts

Statistical profiles show which code is most commonly executed.

Exact measurements show how and why code is executed.
Concepts

Statistical profiles show which code is most commonly executed

Exact measurements show how and why code is executed

XCTests reliably reproduce workloads for profiling
Instruments Templates for Resource Usage
Instruments Templates for Resource Usage

File Activity
Instruments Templates for Resource Usage

File Activity

Network
Instruments Templates for Resource Usage

File Activity

Network

System Trace
Custom Instruments and Templates
Custom Instruments and Templates
Custom Instruments and Templates

- **Counters**: Collects performance counter (PMC) data from running threads on all cores based on regular time intervals.
- **CPU Activity Log**: Helps determine if the energy consumption is related to CPU activity.
- **Disk I/O Latency**: Visualizes the completion times of Disk I/O operations to help diagnose I/O saturation.
- **Disk Usage**: Inspects and profiles an application's interaction with a device storage medium by gathering statistics and visualizing I/O operations.
- **Display**: Records Display events.
- **Energy Log**: Tracks the power requirements of the target device.
- **File Locks**: Observes advisory file locking via the Tock API.
- **Filesystem Activity**: Inspects and profiles an application's filesystem activity, such as its operations on files.
- **Filesystem Suggestions**: Spot filesystem and disk I/O antipatterns that arose during a trace.
- **GCD Performance**: A tool to help spot sub-optimal use of Grand Central Dispatch APIs.
- **GPS On/Off Log**: Tracks when the GPS is enabled.
- **GPU**: Records GPU events.
Summary
Summary

Profile early and often
Summary

Profile early and often

Try out Instruments today
More Information

developer.apple.com/wwdc19/411

Creating Custom Instruments  WWDC 2018

Practical Approaches to Great App Performance  WWDC 2018