What’s New in Energy Debugging

Phillip Azar, Apple/Battery Life
David Choi, Apple/Battery Life
Review general battery life concepts
Review tools for energy debugging
New features for energy debugging
Review general battery life concepts
Review tools for energy debugging
New features for energy debugging
What Is Energy?
What Is Energy?

Power X Time
What Is Energy?

[Diagram showing a graph with 'Power' on the y-axis and 'Time' on the x-axis, depicting a series of peaks and troughs over time.]
What Is Energy?

Diagram showing the relationship between power and time, with energy peaks and valleys.
What Is Energy?
What Is Energy?
What Is Energy?

![Energy Graph]

- **Time**
- **Power**
- **App Active**
- **App Idle**
- **Energy**
What Is Energy?

- App Active
- App Idle
- App Suspended
What Is Energy?

- Power
- Overhead

Overhead

Time
What Is Energy?

Time

Power

Your work

App Active

App Active
What Is Energy?

Time

Power

Your work
Overhead

App Active
Overhead

App Active
Overhead
What Consumes Energy?
What Consumes Energy?

- Processing
- Networking
- Location
- Graphics
Processing
Processing

The “work horse”
Processing

The “work horse”

Workload-dependent
Processing

The “work horse”

Workload-dependent

More operations by your app = more energy
Networking
Networking

Cellular, Wi-Fi, Bluetooth
Networking

Cellular, Wi-Fi, Bluetooth

Traffic-dependent
Networking

Cellular, Wi-Fi, Bluetooth

Traffic-dependent

More network requests = more energy
Location
Location

GPS, WiFi, Cellular
Location

GPS, WiFi, Cellular

Accuracy, frequency-dependent
Location

GPS, WiFi, Cellular

Accuracy, frequency-dependent

More time spent tracking location = more energy
Graphics

Animations and UI
Graphics

Animations and UI

Complexity-dependent
Graphics

Animations and UI

Complexity-dependent

More rendering = more energy
More work = More energy
How Do We Optimize Energy Consumption?
Foreground
Focus on providing value to the user
Foreground

Focus on providing value to the user
• Do required work only
Example
Media app feed
Example
Media app feed

Timer based feed refresh
Example
Media app feed

Timer based feed refresh
Example
Media app feed

Timer based feed refresh
Example
Media app feed

Timer based feed refresh

Feed refresh on demand
- Notification
- User interaction
Example
Media app feed

- Timer based feed refresh
- Feed refresh on demand
  - Notification
  - User interaction
Focus on providing value to the user
• Do work when requested
Focus on **providing value to the user**

- Do work when requested
- Minimize complex UI
Example
Video player app
Example
Video player app
Example

Video player app

Persistent UI controls
Example
Video player app

Persistent UI controls
Example
Video player app

Persistent UI controls
Example
Video player app

Persistent UI controls

Auto dismissed UI controls
- No user interaction
- Timeout based
Example
Video player app

Persistent UI controls

Auto dismissed UI controls
- No user interaction
- Timeout based
Background
Background
Focus on *minimizing workload*
Background

Focus on **minimizing workload**

- Coalesce tasks
Example
App analytics
Example
App analytics

Sending immediately
Example
App analytics

Sending immediately
Example
App analytics

Sending immediately
Example
App analytics

Sending immediately

Sending deferred batches
Example
App analytics

Sending immediately

Sending deferred batches
Background

Focus on **minimizing workload**

- Coalesce tasks
Background

Focus on minimizing workload

• Coalesce tasks
• End tasks quickly
Example

Background modes
Example
Background modes

Lettings tasks expire
Example
Background modes

Lettings tasks expire
Example
Background modes

Lettings tasks expire
Example
Background modes

Lettings tasks expire

Calling completion handlers
- UIBackgroundTask
- VoIP
- Etc
Example
Background modes

Lettings tasks expire

Calling completion handlers
- UIBackgroundTask
- VoIP
- Etc
Processing
Networking
Location
Graphics

Review general battery life concepts
Review tools for energy debugging
New features for energy debugging
Tools for Energy Debugging
Tools for Energy Debugging

Energy Gauges
Tools for Energy Debugging

Energy Gauges
Energy Gauges

Energy Impact

Average Energy Impact

High Energy Impact

Average Component Utilization

Overhead 5%, CPU 15%, GPU 5%, Network 5%

Energy Impact

Overhead

Network Activity

Location Activity

High CPU Utilization

Network Activity

Location Activity

High GPU Utilization
Energy Report

Energy

Average Energy Impact

High
Energy Impact

Average Component Utilization

- Overhead 54.1%
- CPU 38.8%
- Network 17%
- Location 0%
- GPU 0%

Energy Impact

00:00:000 00:10:000 00:15:000 00:20:000 00:25:000 00:30:000 00:35:000 00:40:000

Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
CPU usage of greater than 20%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locating uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
Excessive rendering and animations reduce performance and put system resources out of low-power states as expected these days.
**Energy**

**Average Energy Impact**

- **High**

**Energy Impact**

- **Overhead**
- **CPU**
- **Network**
- **Location**
- **GPU**

**Average Component Utilization**

- Overhead: 54.1%
- CPU: 28.8%
- Network: 17%

**Overhead**

Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

**High CPU Utilization**

CPU usage greater than 25%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

**Network Activity**

Network activity occurs in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

**Location Activity**

Location activity performed by your app. More precise and frequent location use more energy. Request location and increase precision only when truly necessary.

**High GPU Utilization**

Extreme animation and graphics-intensive elements and activities can run out of free energy quickly. Reduce them from the app.
**Energy Impact**

- **Overhead**
  Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

- **High CPU Utilization**
  CPU usage of greater than 25%. High CPU utilization rapidly drains a device’s battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

- **Network Activity**
  Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

- **Location Activity**
  Location activity performed by your app. More precise and frequent locationing uses more energy. Request location and increase precision only when truly necessary.

- **High GPU Utilization**
  Extreme avoidant and animations, making use of touchscreen and full screen capture out of the normal usage or request them form.
Average Energy Impact

High Energy Impact

Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
Cpu usage of greater than 25%. High CPU utilization rapidly drains a device’s battery. Always use the CPU efficiently and return to idler as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locationing uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
External animations and graphics activate performance levels and will consume energy out of the power budget if not trimmed.
Energy

Average Energy Impact

High
Energy Impact

Average Component Utilization

- Overhead: 54.1%
- CPU: 28.6%
- Network: 17%
- Location: 0%

Energy Impact

Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
CPU usage of greater than 25%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locationing uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
Extreme visual effects and animations adding unnecessary and often unseen content out of the user's visual field slow down theiran performance.
Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
CPU usage of greater than 25%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locating uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
Excessive rendering and animations, setting textures, or not managing resources out of low power states can impact them.
Energy

Average Energy Impact

High

Energy Impact

Average Component Utilization

Overhead 64.1%
CPU 26.4%
Network 17%

Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
CPU use of greater than 25%. High CPU utilization rapidly drains a device’s battery. Always use the CPU efficiently and return to Idle as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for protracted periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locating uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
Excessive rendering and animations can drain power. Always turn off unnecessary rendering and cut down on excessive use of the screen as much as possible.
High CPU Utilization

CPU usage is greater than 25%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

Network Activity

Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity

Location activity performed by your app. More precise and frequent locationing uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization

Excessive animations and graphics settings increase power consumption and can drain your device's battery even more than if you perform these tasks yourself.
Energy Gauges Are Great for...
Energy Gauges Are Great for...

High level energy characterization

Rapid profiling
### Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

### High CPU Utilization
CPU usage of greater than 20%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

### Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

### Location Activity
Location activity performed by your app. More precise and frequent locating uses more energy. Request location and increase precision only when truly necessary.

### High GPU Utilization
Graphics activity requested by your app. Extraneous graphics and animations reduce responsiveness and pull system resources out of low-power states or prevent them from powering down all together, resulting in significant energy use. Make updates to visible content only, reduce the use of opacity, and prefer lower, consistent frame rates when performing animations.

### Background State
Your app is in a background state, keeping the system awake. Even an idle background app uses energy. If your app requires background operations, use deferral APIs that let the system schedule the work efficiently and wake to run your app only when necessary. Otherwise, reduce activity immediately when placed in the background and notify the system once the activity is complete.

### Foreground State
Your app is in the foreground. Use recommended APIs, batch and reduce network operations, and avoid unnecessary updates to the user interface. Strive to make your app absolutely idle when it's not responding to user input.
Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
CPU usage of greater than 20%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locating uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
Graphics activity requested by your app. Extraneous graphics and animations reduce responsiveness and pull system resources out of low-power states or prevent them from powering down all together, resulting in significant energy use. Make updates to visible content only, reduce the use of opacity, and prefer lower, consistent frame rates when performing animations.

Background State
Your app is in a background state, keeping the system awake. Even an idle background app uses energy. If your app requires background operations, use deferral APIs that let the system schedule the work efficiently and wake to run your app only when necessary. Otherwise, reduce activity immediately when placed in the background and notify the system once the activity is complete.

Foreground State
Your app is in the foreground. Use recommended APIs, batch and reduce network operations, and avoid unnecessary updates to the user interface. Strive to make your app absolutely idle when it’s not responding to user input.
Overhead
Overhead represents energy use as a result of bringing up radios and other system resources your app needs to perform work.

High CPU Utilization
CPU usage of greater than 20%. High CPU utilization rapidly drains a device's battery. Always use the CPU efficiently and return to idle as quickly as possible when not directly responding to user input.

Network Activity
Network activity occurring in response to your app. Networking brings up radios, which require power for prolonged periods. Batch network activity whenever possible to reduce overhead.

Location Activity
Location activity performed by your app. More precise and frequent locating uses more energy. Request location and increase precision only when truly necessary.

High GPU Utilization
Graphics activity requested by your app. Extraneous graphics and animations reduce responsiveness and pull system resources out of low-power states or prevent them from powering down all together, resulting in significant energy use. Make updates to visible content only, reduce the use of opacity, and prefer lower, consistent frame rates when performing animations.

Background State
Your app is in a background state, keeping the system awake. Even an idle background app uses energy. If your app requires background operations, use deferral APIs that let the system schedule the work efficiently and wake to run your app only when necessary. Otherwise, reduce activity immediately when placed in the background and notify the system once the activity is complete.

Foreground State
Your app is in the foreground. Use recommended APIs, batch and reduce network operations, and avoid unnecessary updates to the user interface. Strive to make your app absolutely idle when it’s not responding to user input.
Other Profiling Templates and Instruments
Other Profiling Templates and Instruments

Energy diagnostics profiling template
Activity monitor profiling template
Core animation profiling template
GPU driver profiling template
Location energy instrument
Network profiling template
Instruments Are Great for...
Instruments Are Great for...

- Root cause analysis
- In depth profiling
Instruments Are Great for...

- Root cause analysis
- In depth profiling
- Untethered profiling
Working Demo
Energy Debugging
Working Demo

Takeaways

Energy Gauges for rapid iteration

Instruments for root cause analysis

Make energy a top priority
You’ve shipped your app.
Whats next?
Review general battery life concepts
Review tools for energy debugging
New features for energy debugging
New Features for Energy Debugging

David Choi, Battery Life
Now that You’ve Shipped Your App...

How do you know if your customers are experiencing energy issues?

How do you debug a customer energy issue?
A New Way of Debugging Energy Issues

Xcode Energy Logs
Xcode Energy Organizer
Xcode Energy Logs

Reports high CPU energy events
Points out energy hotspots in code
Data from TestFlight and App Store
When Are Xcode Energy Logs Generated?

High CPU Energy Events
- 80% CPU spin over 3 minutes in foreground
- 80% CPU spin over 1 minute in background
When Are Xcode Energy Logs Generated?

- **High CPU Energy Events**
  - 80% CPU spin over 3 minutes in foreground
  - 80% CPU spin over 1 minute in background

[Graph showing CPU usage over time with 80% threshold]
Xcode Energy Logs: Impact on User Battery Life
Up to 1% battery drop
Xcode Energy Logs: Impact on User Battery Life

Up to 1% battery drop

- 8 minutes of talk time
- 6 minutes of browsing
- 30 minutes of music
What’s in an Energy Log?

Energy condition that triggered the report

Device type and app build number

Weighted call graph that shows energy hotspots
Xcode Energy Log: Weighted Call Graph
Xcode Energy Log: Weighted Call Graph

main()

method1()

method1()

method3()

method2()

method4()

method2()

method3()

High CPU Energy Event detected!
Xcode Energy Log: Weighted Call Graph

main()

method1()

method1()

method3()

method2()

method4()

method2()

method3()
Xcode Energy Log: Weighted Call Graph

main()
method1()
method1()
method3()
method2()
method4()
main()
method1()
method2()
main() method1() method3() method2() method3() method2() method3() method2() method3() method2() method3() method2() method3() method2() method3() method2() method3() method2() method3() method2() method3() method2() method3()
Xcode Energy Log: Weighted Call Graph

- main()
  - method1()
    - method2()
    - method3()
  - method1()
  - method2()
    - method3()
  - method1()
    - method4()
Weighted Call Graph

Collection of backtrace samples

Aggregated by sample counts

More samples mean heavily executed code

6 main()
   5 method1()
   3 method2()
   1 method3()
   1 method3()
   1 method4()
How to Access Energy Logs?
Xcode Energy Organizer
Xcode Energy Organizer

Test Flight and App Store iOS apps
Xcode Energy Organizer

Test Flight and App Store iOS apps

Recent statistics
Xcode Energy Organizer

Test Flight and App Store iOS apps

Recent statistics

List of top issues
Xcode Energy Organizer

Test Flight and App Store iOS apps

Recent statistics

List of top issues

Weighted call graph
Xcode Energy Organizer

Test Flight and App Store iOS apps
Recent statistics
List of top issues
Weighted call graph
Page through logs
Xcode Energy Organizer

Test Flight and App Store iOS apps
Recent statistics
List of top issues
Weighted call graph
Page through logs
Open in Project
Demo

Xcode Energy Organizer
Energy Organizer
Demo takeaways

Discover top energy issues from the field

View energy hotspots with weighted call graphs

“Open in Project” to debug code
Summary

Think about energy use in design

Use energy gauges and instruments

Explore the new Xcode Energy Organizer
More Information


Power and Performance Lab

Technology Lab 9

Friday 9:00AM