Developing for WatchKit 1.0

(Starting at 2:20)

Developing for WatchKit 1.0

Curt Clifton
@curtclifton
curtclifton.net

The Omni Group

Outline

- Watch app parts

- Conceptual model

- Syncing data to Apple Watch

- Debugging Apple Watch apps
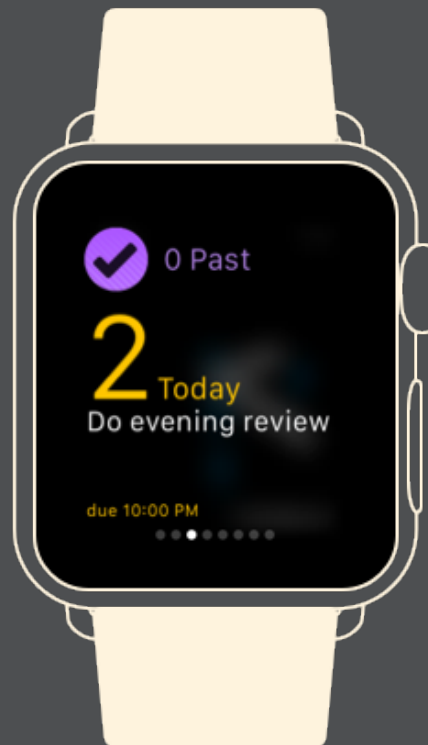
- Some challenges

Watch App Parts

Your Watch app can include notifications, glances, the main app, and handoff

Your Watch app can include notifications, glances, the main app, and handoff

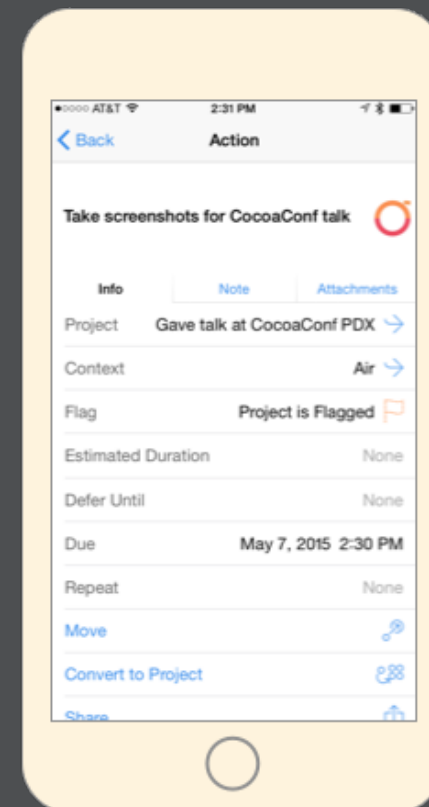# Your Watch app can include notifications, glances, the main app, and handoff

Your Watch app can include notifications, glances, the main app, and handoff

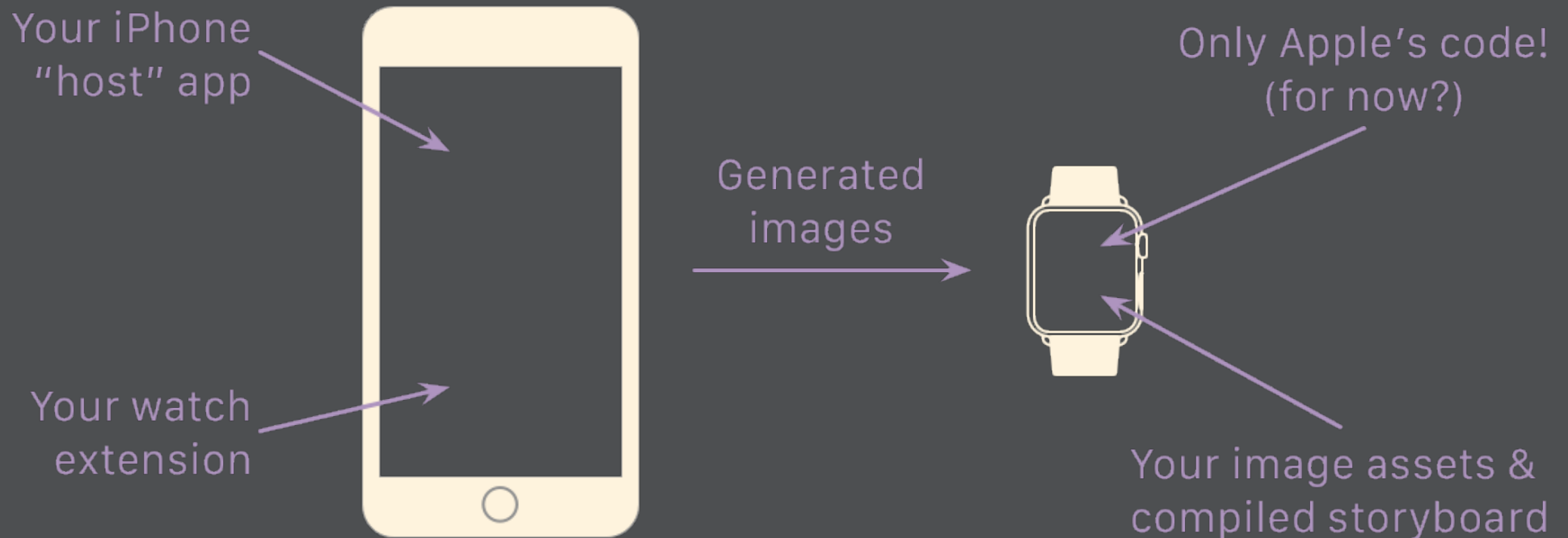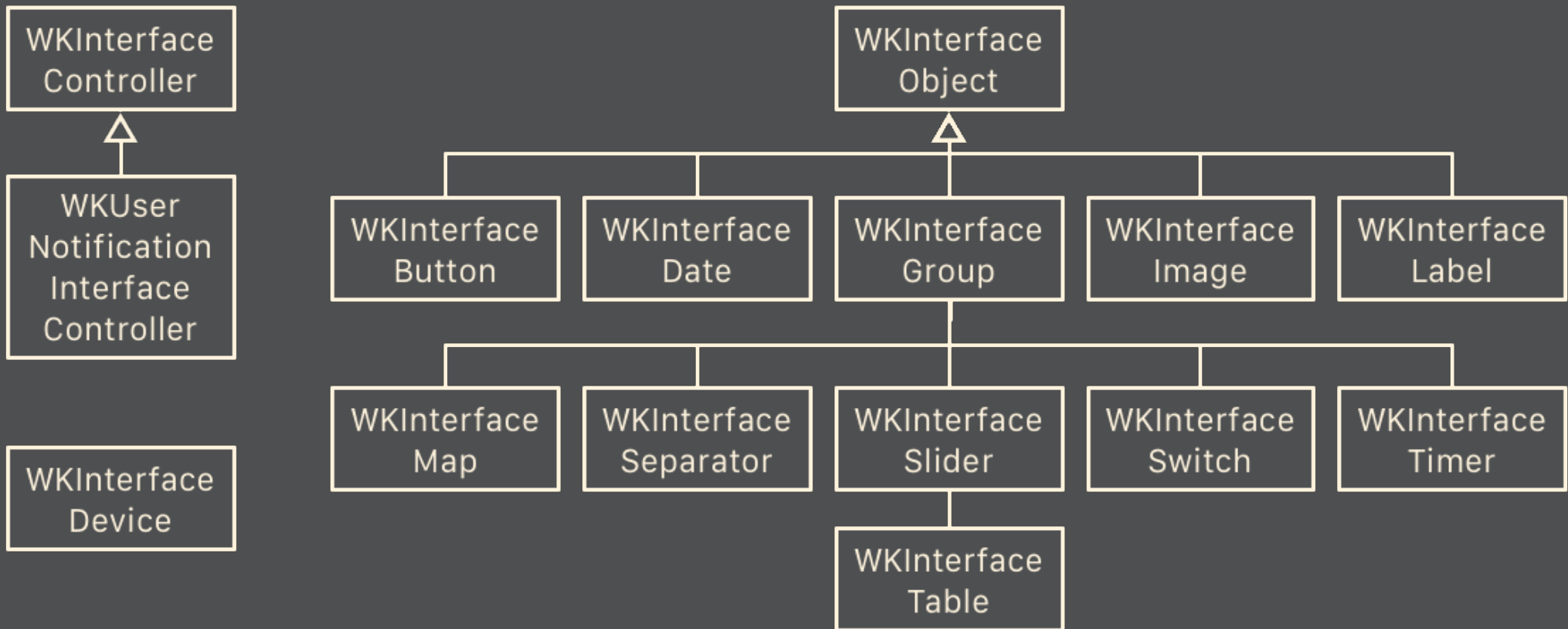# Your Watch app can include notifications, glances, the main app, and handoff

Conceptual Model

In WatchKit 1.0 your code runs in an extension on the iPhone

In WatchKit 1.0 your code **runs** in an extension **on the iPhone**

Your iPhone "host" app

Your watch extension

Generated images

Only Apple's code! (for now?)

Your image assets & compiled storyboard

All the WatchKit classes fit on one slide that you can read
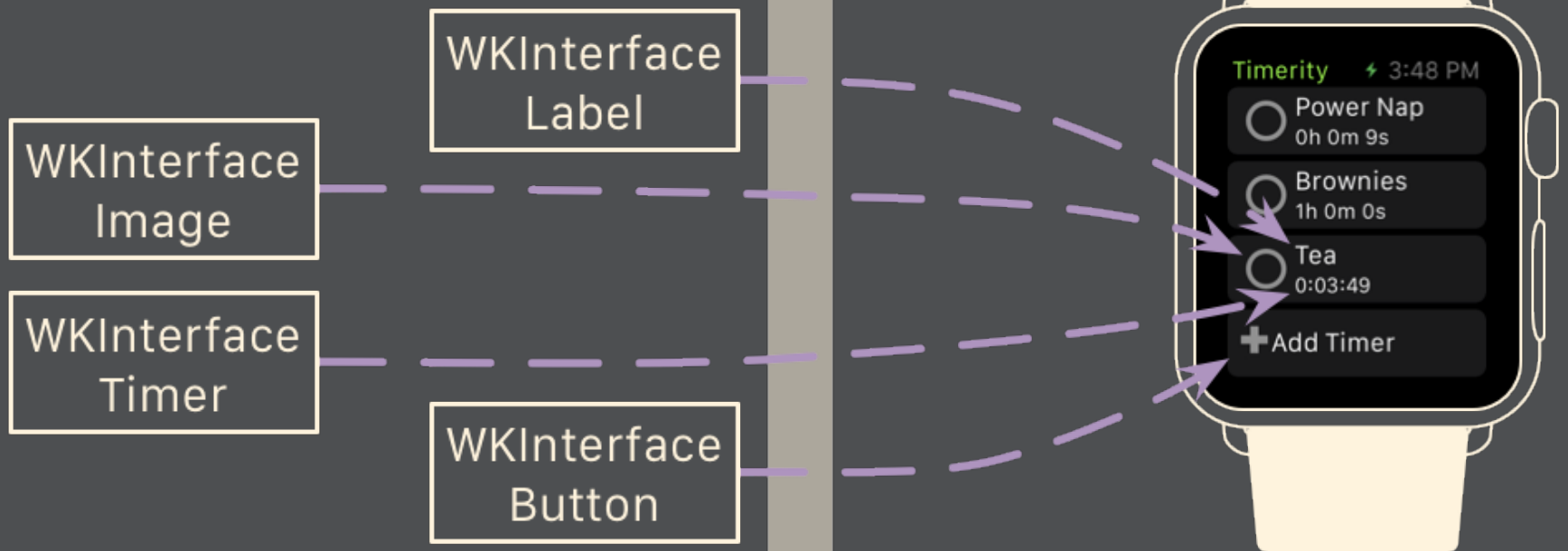
All the WatchKit classes fit on one slide that you can read

All the WKInterfaceObjects are proxies for views on the watch

All the WKInterfaceObjects are proxies for views on the watch

Your watch extension on iPhone

WKInterface Label

WKInterface Image

WKInterface Timer

WKInterface Button

Timerity ⚡ 3:48 PM

○ Power Nap
0h 0m 9s

○ Brownies
1h 0m 0s

○ Tea
0:03:49

➕ Add Timer

All the WKInterfaceObjects are proxies for views on the watch

Your watch extension on iPhone

interfaceProperty
-setHidden:
-setAlpha:
-setWidth:
-setHeight:
-setAccessibility...:
-setText:
-setTextColor:
-setAttributedText:

WKInterface Label

WKInterface Button

Timerity ⚡ 3:48 PM

○ Power Nap
0h 0m 9s

○ Brownies
1h 0m 0s

○ Tea
0:03:49

➕ Add Timer

Conceptual Model

- In WatchKit 1.0 your code **runs** in an extension **on the iPhone**

- All the WKInterfaceObjects are proxies for views on the watch

Sample App

Syncing Data to Apple Watch

Your watch extension is a separate process from your iPhone app—you just signed up for data syncing

Watch apps are killed mercilessly
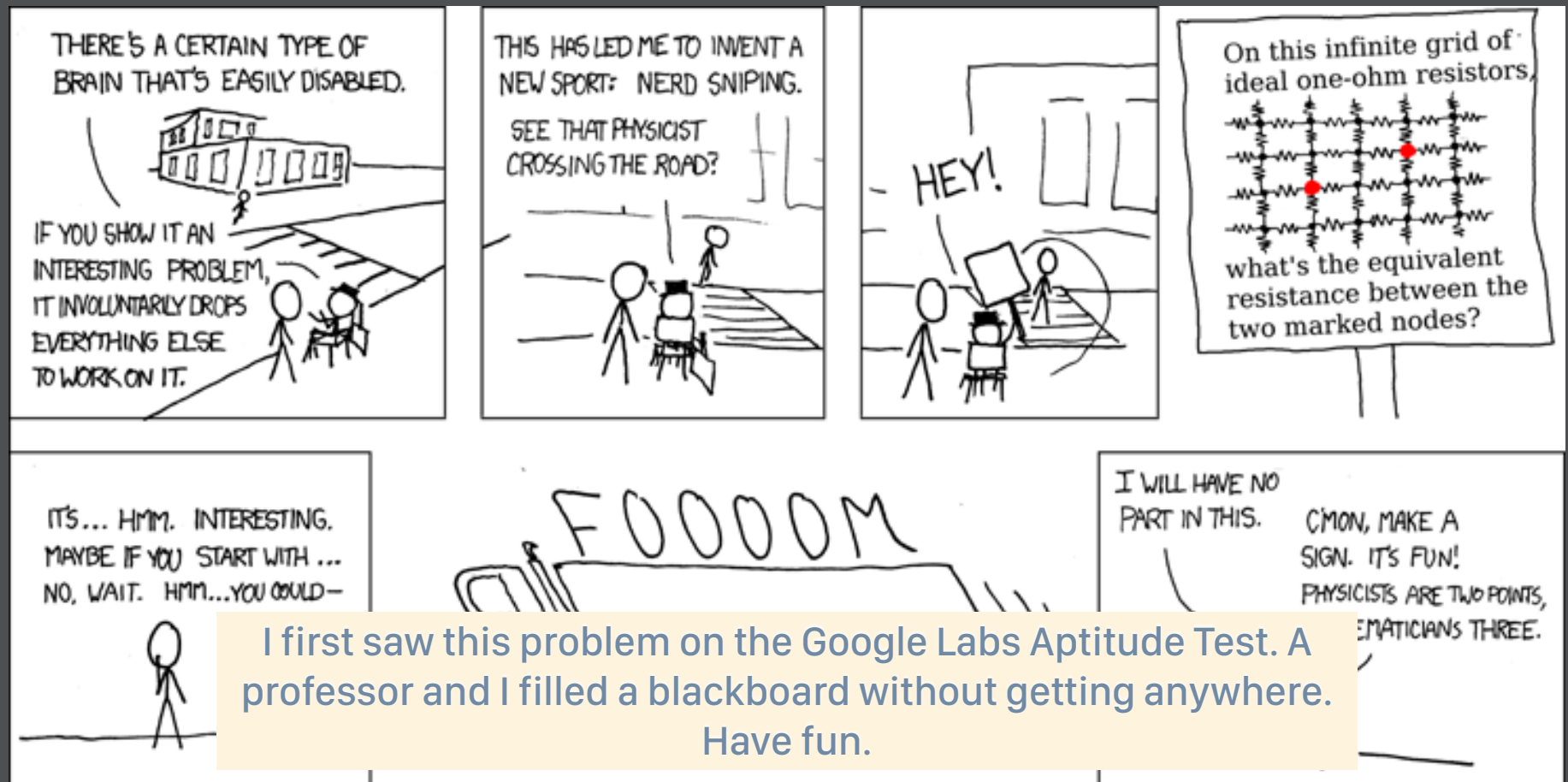
Network operations are right out

Monty Python and the Holy Grail

What's a developer to do?

- File coordination? <u>Nope.</u>

- <u>App Group entitlement and user defaults</u>

- <u>Shared CoreData database</u>

- Shared SQLite database

- Darwin Notifications—<u>MMWormhole</u>

- Seed file & callbacks
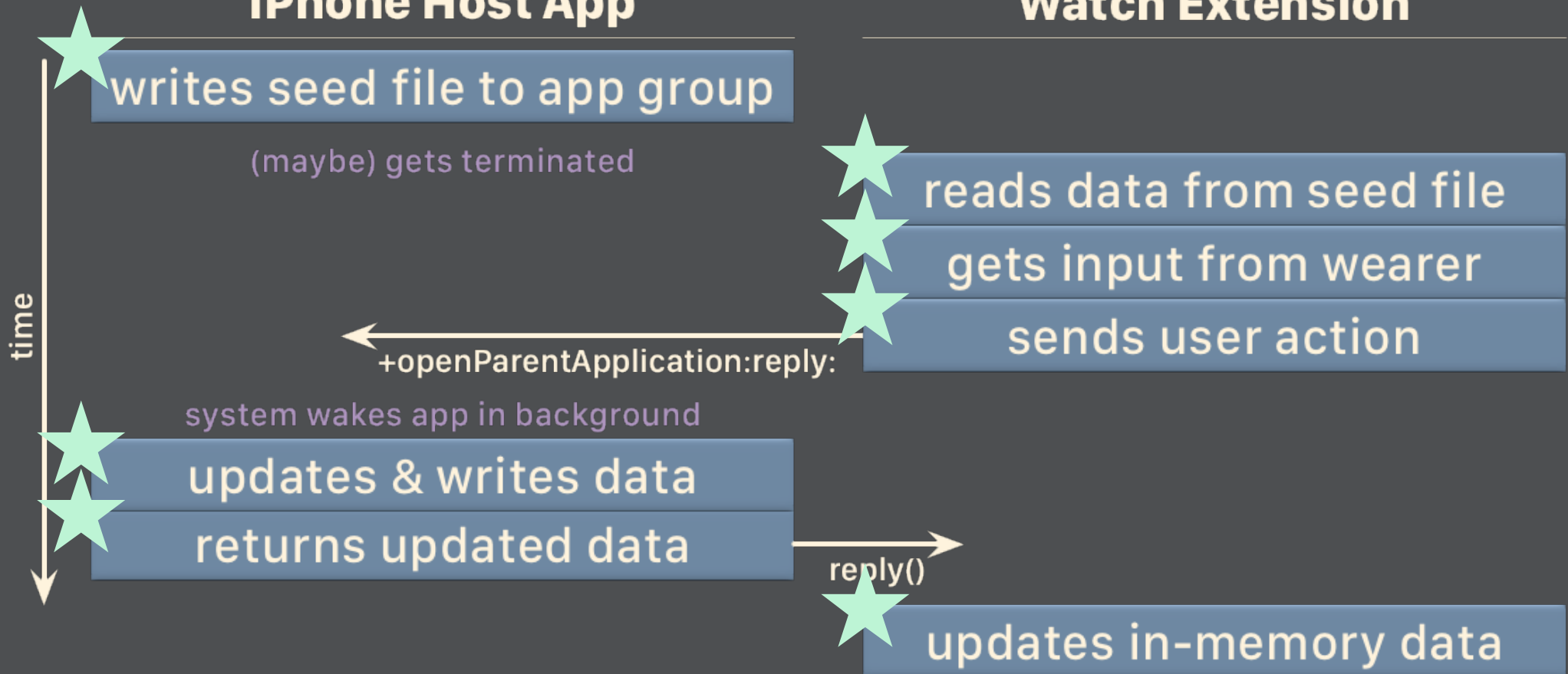
Cartoon of the Day

I first saw this problem on the Google Labs Aptitude Test. A professor and I filled a blackboard without getting anywhere. Have fun.

Syncing Using Seed File & Callbacks

iPhone Host App   **Watch Extension**

writes seed file to app group

(maybe) gets terminated

reads data from seed file

gets input from wearer

sends user action

+openParentApplication:reply:

system wakes app in background

updates & writes data

returns updated data

reply()

updates in-memory data

time

Syncing Data to Apple Watch

- Your watch extension is a separate process from your iPhone app—you just signed up for data syncing

- Watch apps are killed mercilessly—network operations are right out

Debugging Apple Watch Apps

- Run iPhone app, open Watch "external display"

- Launch the WatchKit app from Xcode

- Launch the iPhone app from Springboard

- In debugger, connect to the iPhone app

- Use Console to watch Simulator's system log
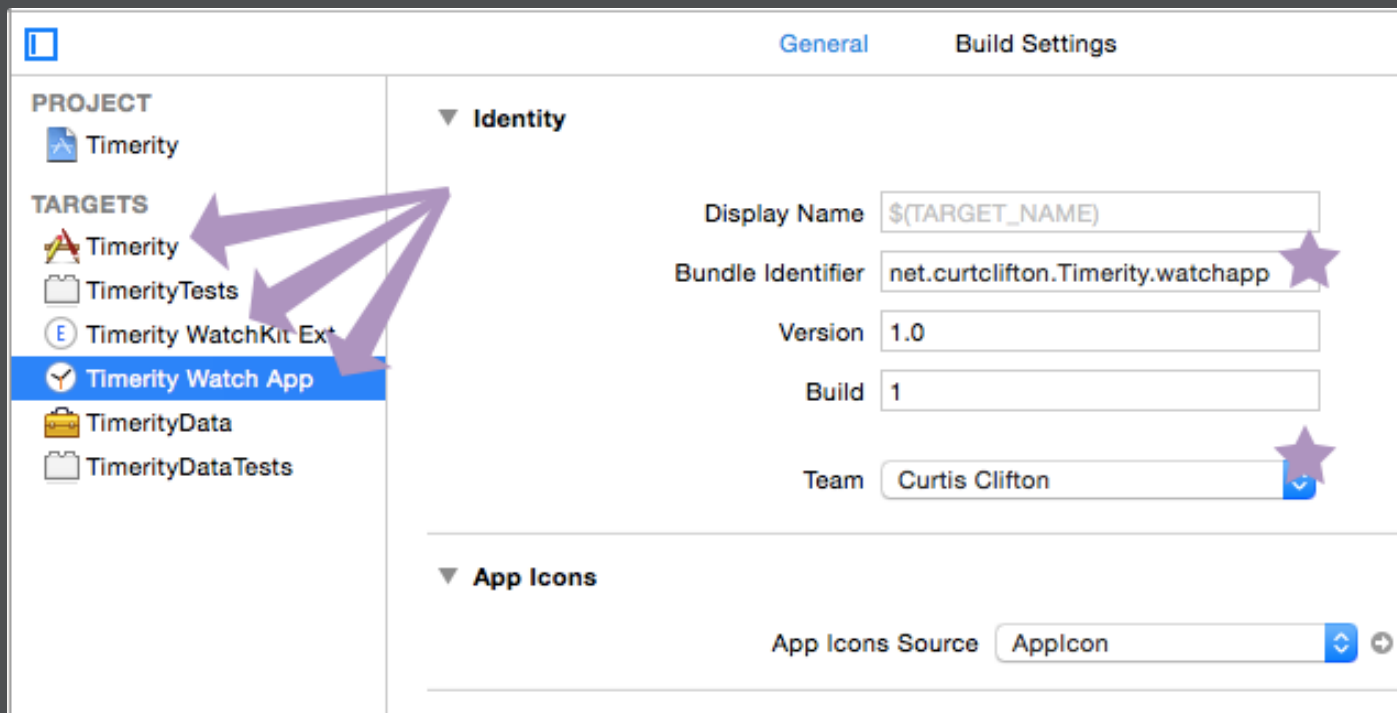
  - *…/data/Library/Logs/system.log*

Challenges

Provisioning

What fresh circle is this?

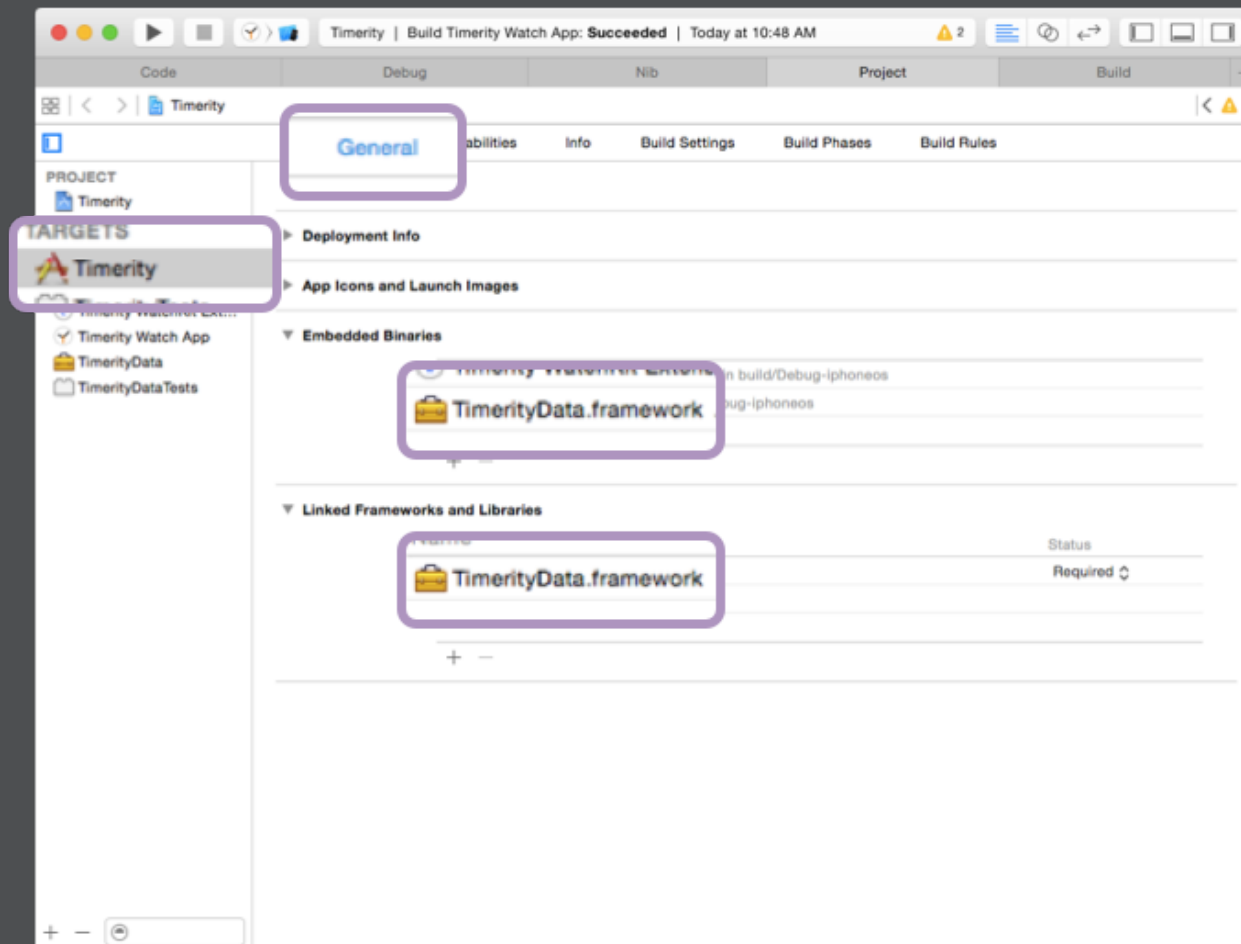# You need to provision your Apple Watch and configure (at least) three targets

WKInterface objects silently ignore commands if their proxied views aren't active
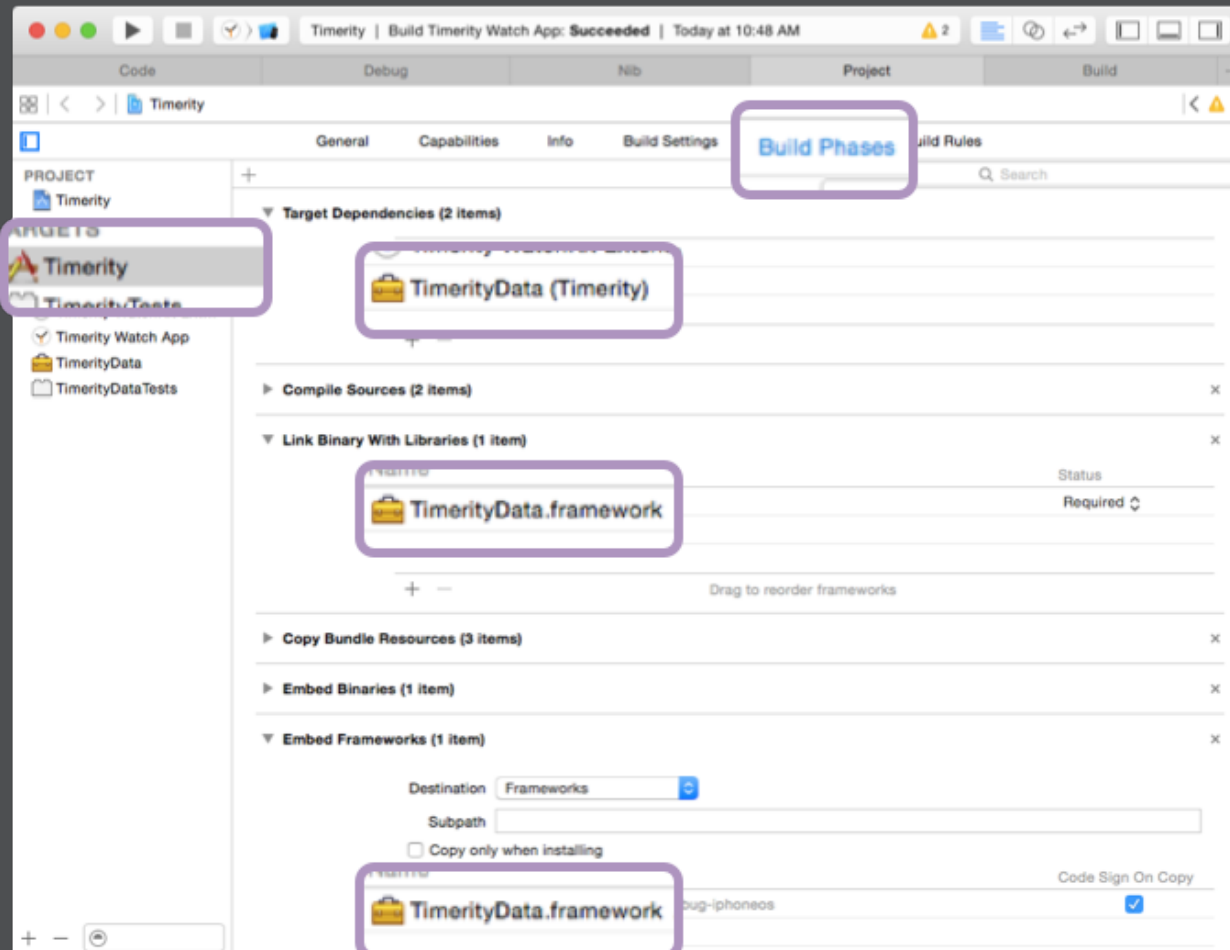
Shared frameworks help encapsulate your data model but require some initial configuration work
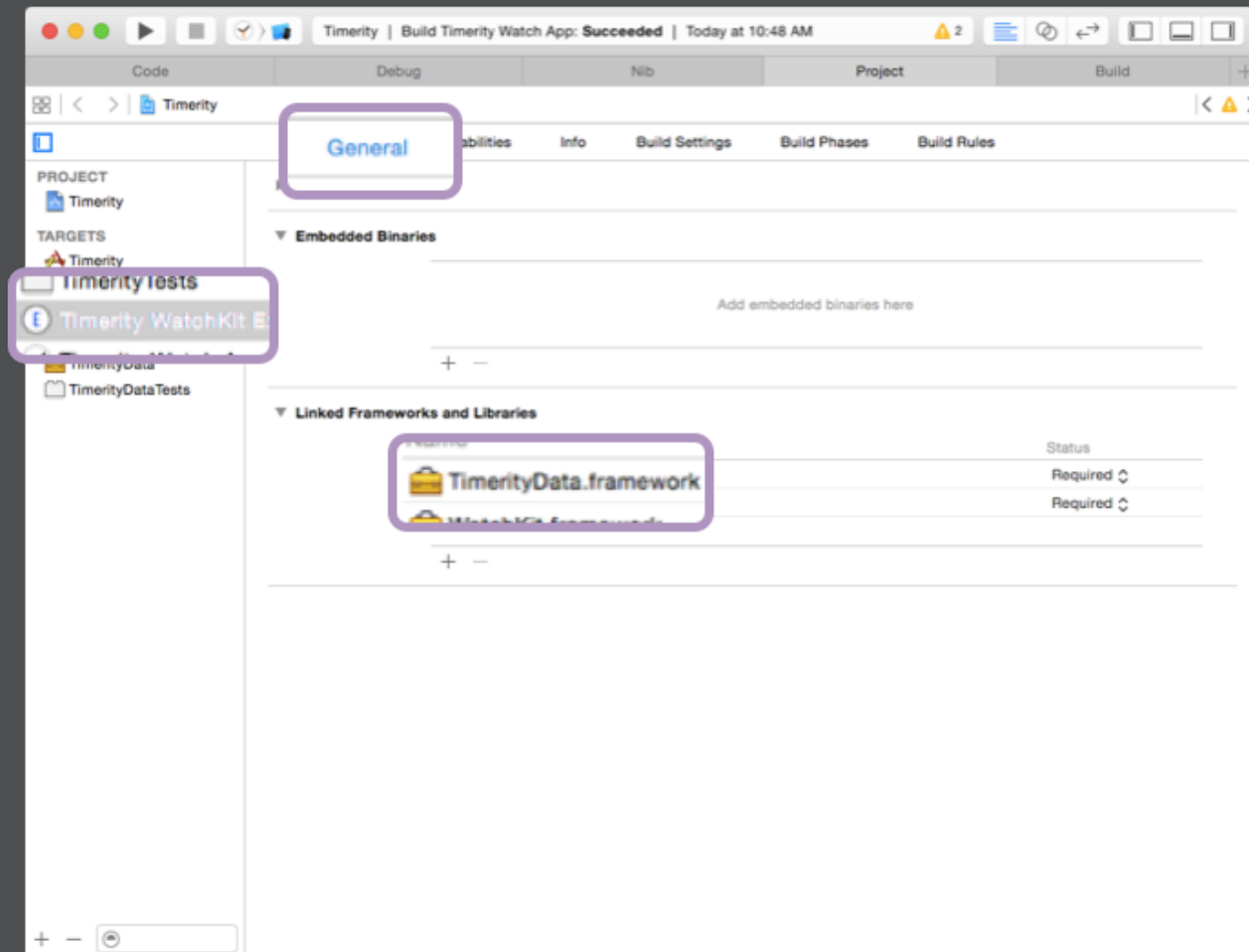
Shared frameworks help encapsulate your data model but require some initial configuration work

Shared frameworks help encapsulate your data model but require some initial configuration work

Shared frameworks help encapsulate your data model but require some initial configuration work

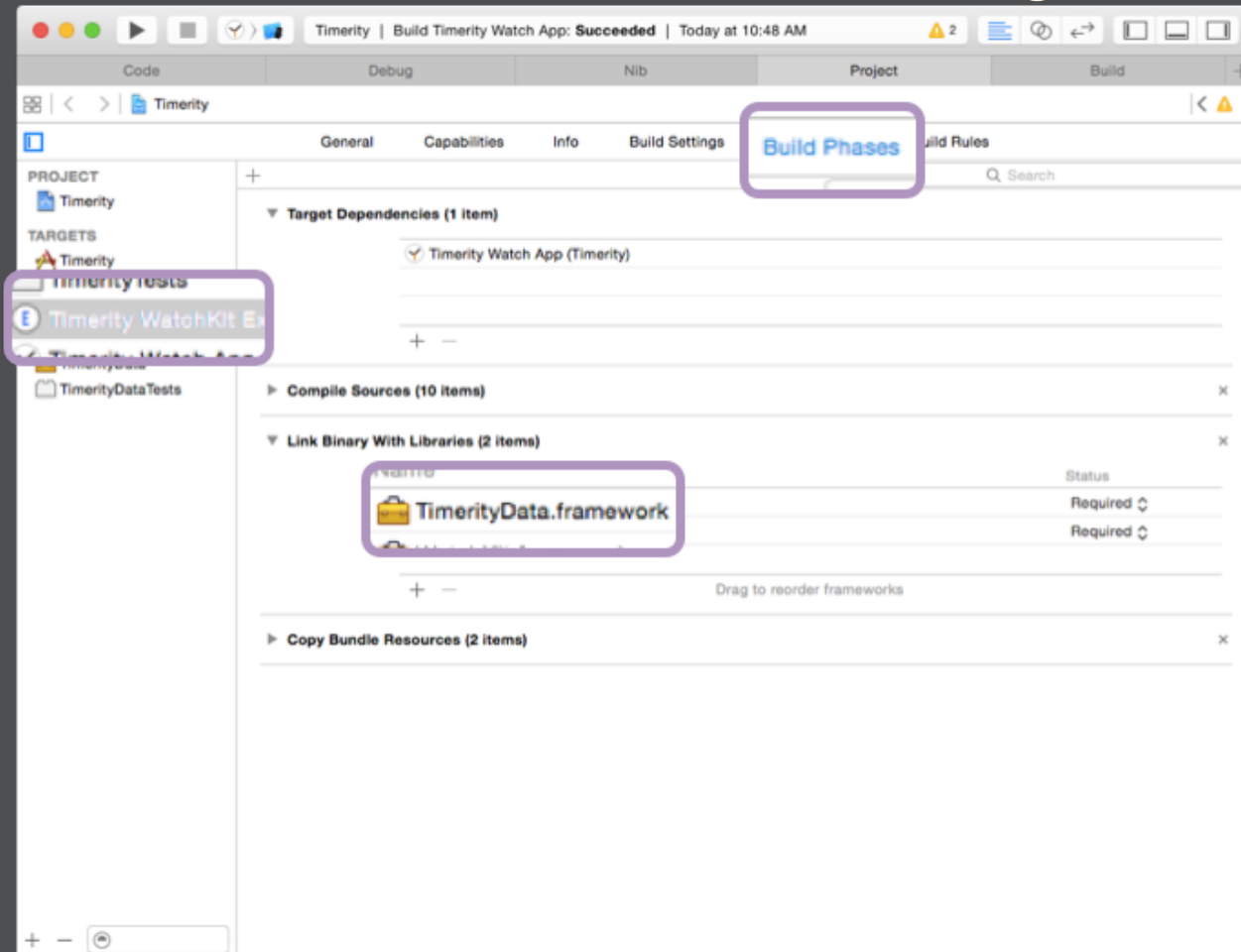# Shared frameworks help encapsulate your data model but require some initial configuration work

Shared frameworks help encapsulate your data model but require some initial configuration work

public

Useful Resources

From Apple

- Dev and design resources

- Human Interface Guidelines

- Marketing site

- "Secret" watch kit tips

From Others

- "As I Learn WatchKit" — David Smith
  - "Thinking about Load Time"
- "Architecting Your App for WatchKit" — Natasha Murashev
- Apple Watch Code Signing — Jared Sinclair
- WatchKit Development Tips — Mike Swanson

Thanks!

@curtclifton
curtclifton.net
https://github.com/curtclifton/timerity

The Omni Group