

AsyncDisplayKit

1. Giới thiệu

AsyncDisplayKit (ASDK) là một thư viện được sử dụng lần đầu tiên bởi ứng dụng Facebook paper nhằm tối ưu hóa khả năng hiển thị nhờ vào việc tận dụng CPU nhiều core trên các thiết bị mới (> iphone 5s). ASDK giúp cho app có thể render ở mức 60fps nhờ vào việc đưa một cách triệt để các tác vụ render trên main thread xuống dưới background thread, nhờ đó main thread có nhiều thời gian hơn để có thể đáp ứng các thao tác của người dùng một cách nhanh nhất có thể.

Các hàm API của ASDK được team phát triển xây dựng một cách giống nhất có thể so với UIKit. Nhờ đó người mới tiếp cận bộ thư viện này có thể nhanh chóng làm quen.

2. Cách ASDK tối ưu hóa nhờ vào multithread

Thông thường, các tác vụ liên quan đến UI đều được xử lý bởi main thread. Các tác vụ đó có thể là:

- Render các thành phần giao diện (Bo góc, đổ bóng, vẽ UI elements,...).
- Giải mã các hình ảnh.
- Xử lý các tương tác của người dùng (chạm, vuốt, ...).

Một giao diện được xem là mượt mà khi nó được render ở mức 60fps, do đó main thread buộc phải xử lý tất cả các tác vụ trên trong thời gian rất ngắn. Đối với các giao diện phức tạp, việc render sẽ mất nhiều thời gian hơn, dẫn đến việc frame drop làm cho giao diện trở nên giật, lag, gây giảm trải nghiệm người dùng.

Để xử lý vấn đề này, ASDK đã đưa hầu hết các tác vụ render xuống dưới background thread, từ đó main thread có nhiều thời gian hơn để có thể đáp ứng tốt cho người dùng, đảm bảo UI được render ở mức 60fps một cách liên tục.

Để làm được điều này thì ASDK đã viết lại (hoặc wrap) hầu hết các UI Elements của UIKit để giúp chúng hỗ trợ multithread, thread safe. Song song với đó ASDK còn tạo ra cách khác để có thể dựng giao diện một cách tốt hơn và hiệu quả hơn so với cách dựng giao diện truyền thống bằng auto layout hay code thuần. Do nếu dùng auto layout, thiết bị phải tốn rất nhiều tài nguyên để có thể giải các ràng buộc về mặt giao diện để có thể render ra giao diện hoàn chỉnh. Do đó gánh nặng gây ra trên main thread do auto layout là rất lớn nên ASDK đã đưa ra giải pháp của họ để có thể thực hiện việc render ở dưới background thread.

3. Intelligent Preloading

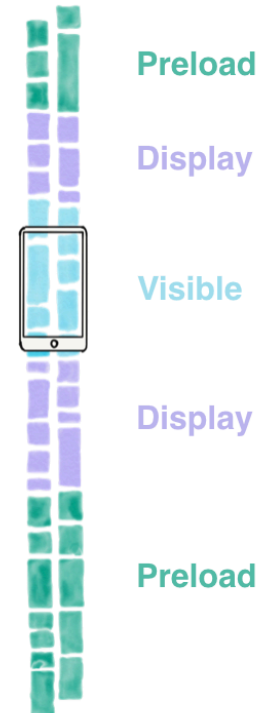
Đây là thứ giúp tối ưu hóa khả năng đa nhiệm của con chip đa nhân trên các thiết bị iOS. Giúp các thao tác cuộn trên TableView, CollectionView hay ScrollView được mượt mà nhất có thể.

Như hình bên, nội dung scrollable sẽ được chia ra làm 3 phần:

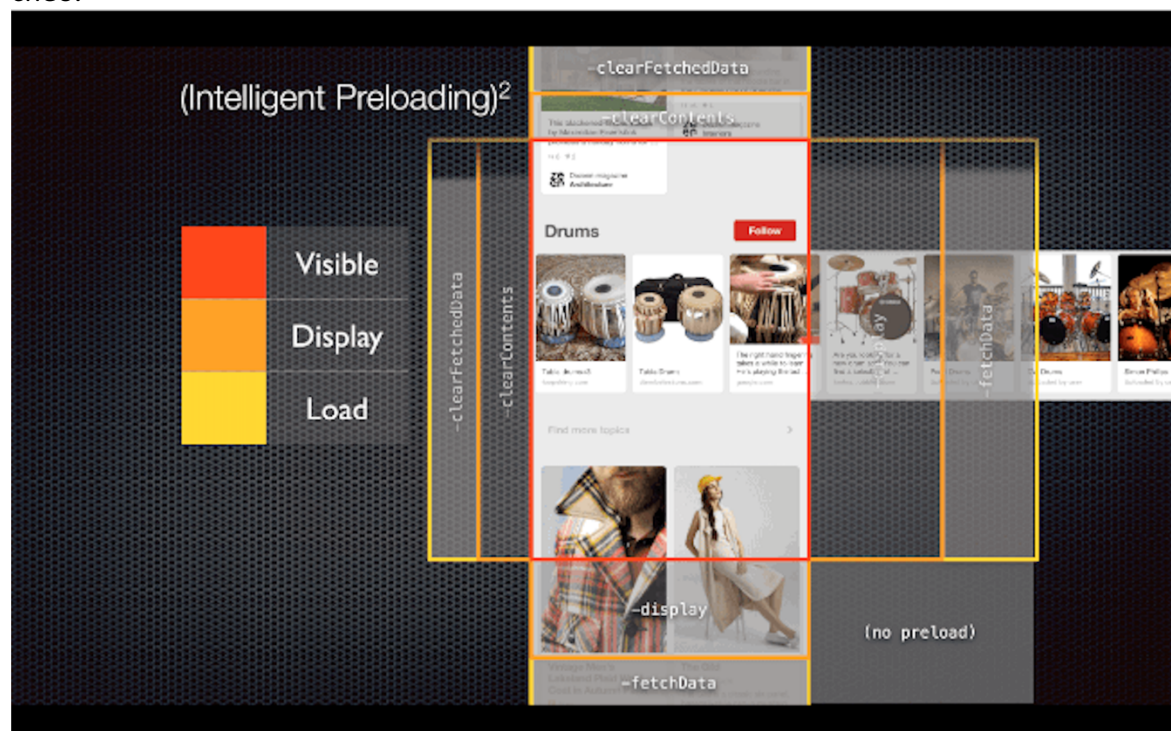
- **Preload:** Khi các item vào vùng này, nó sẽ được ASDK yêu cầu thực hiện các tác vụ khởi tạo, lấy dữ liệu (model), download các hình ảnh từ trên mạng...
- **Display:** Khi này, các item sẽ tiến hành việc render, decode image... dưới background thread sau khi đã đủ các dữ liệu cần thiết.
- **Visible:** Khi này ít nhất 1 pixel của item đã được hiển thị.

Nếu như thao tác cuộn của người dùng quá nhanh dẫn đến việc trình tự trên không được hoàn thành. Một placeholder sẽ được ASDK hiển thị thay thế. Ngay khi item hoàn thành trình tự trên dưới background thread, nó sẽ được cập nhật lại.

Mặc định, kích thước của vùng này sẽ bằng nhau và bằng đúng kích thước của màn hình. ASDK cũng cho phép chúng ta có thể tùy chỉnh lại điều này.



Kể từ phiên bản ASDK 2.0, Intelligent Preloading còn thông minh hơn nữa khi có thể hoạt động một cách đa chiều. Tức không chỉ chiều thẳng đứng, mà còn có thể là ngang hoặc chéo.



Ví dụ scrollView (SR1) theo chiều thẳng đứng, nhưng nó có chứa thêm một scrollView (SR2) theo chiều ngang. Lúc này khi chúng ta scrolling SR1 theo chiều dọc, nếu SR2 scroll tới vùng visible của SR1 thì cơ chế Intelligent Preloading của SR2 sẽ được kích hoạt. Còn nếu không, cơ chế preload cho SR2 sẽ mặc định được tắt đi.

4. Các tiện ích được ASDK cung cấp.

Đơn vị nhỏ nhất của ASDK là node (tương đương với UIView), đây là class wrapper của các UI Element trong UIKit.

Tất cả các node của ASDK đều được viết ra để hỗ trợ multithread và Intelligent Preloading như kể trên. Khi ta sử dụng chúng bên trong các Node Container, thì các cơ chế kể trên sẽ tự động được ASDK quản lý, ta không cần phải làm gì thêm.

a. ASNetworkImageNode

Đây là class dùng để hiển thị hình ảnh tương tự như UIImageView. Nhưng đặc biệt hơn là nó có thể hỗ trợ thêm auto download hình ảnh từ mạng về và hỗ trợ luôn việc caching.

Ngoài ra, class này còn hỗ trợ hiển thị hình ảnh GIF.

b. ASVideoNode

Đây là class giúp hiển thị và chơi các video