



Flutter 入門篇

以 CNOC 障礙通報 App 為例

Acer AEB
Joe Yu
2021.02.04
V2.0



Flutter

- UI SDK
 - Google 開發
 - Open Source
 - 跨平台
- 開發語言：Dart



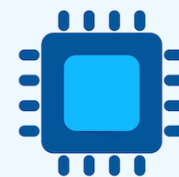
Mobile



Web



Desktop



Embedded

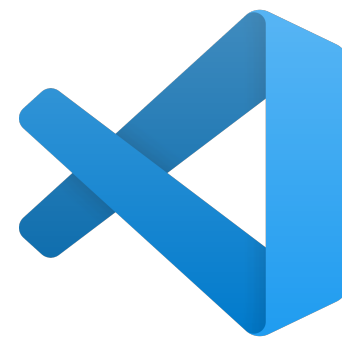
選擇

Why Choose Flutter

- UI：可擴展、可重用、隨便拉拉就有個樣子
- 效能：編譯成原生的 machine code、Up to 120 fps
- 快速開發：小團隊、Single Codebase、Hot Reload
- 官方支援：packages support、迭代速度

開發環境 IDE

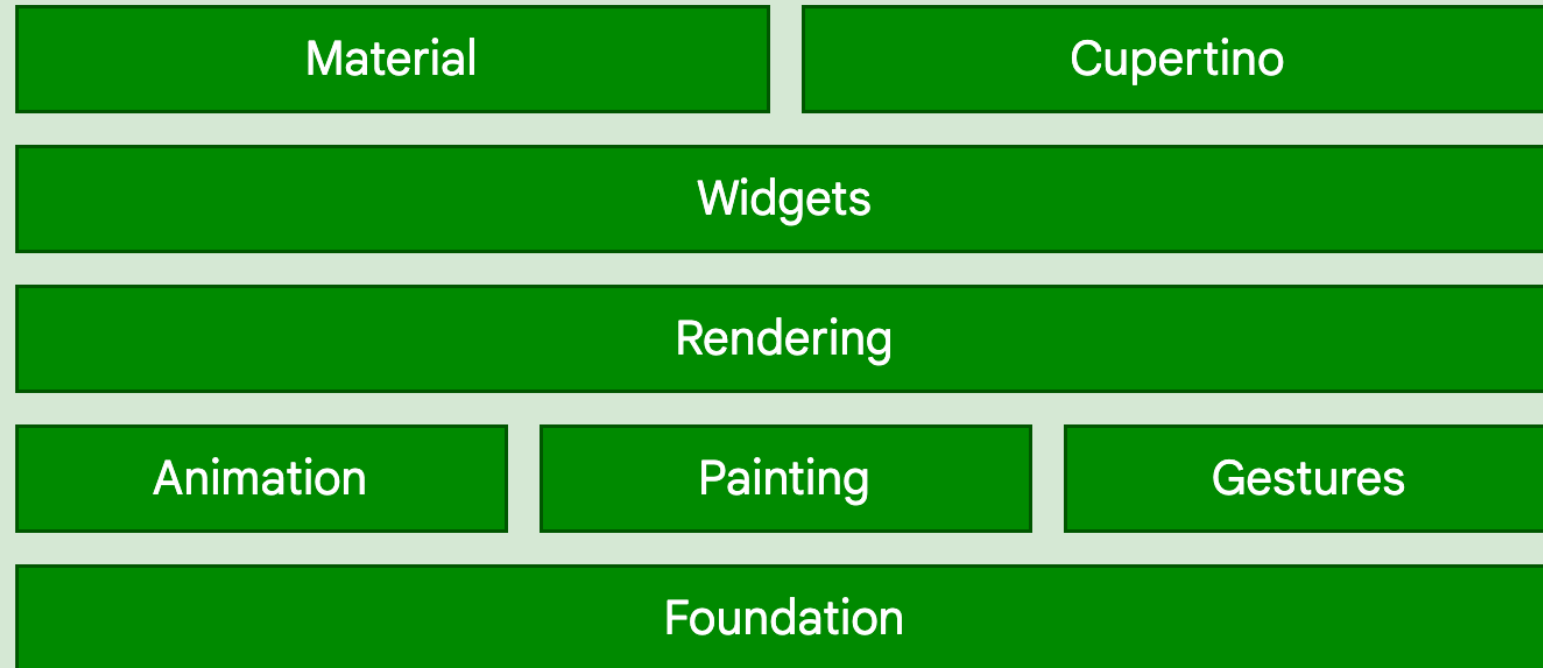
- Visual Studio Code



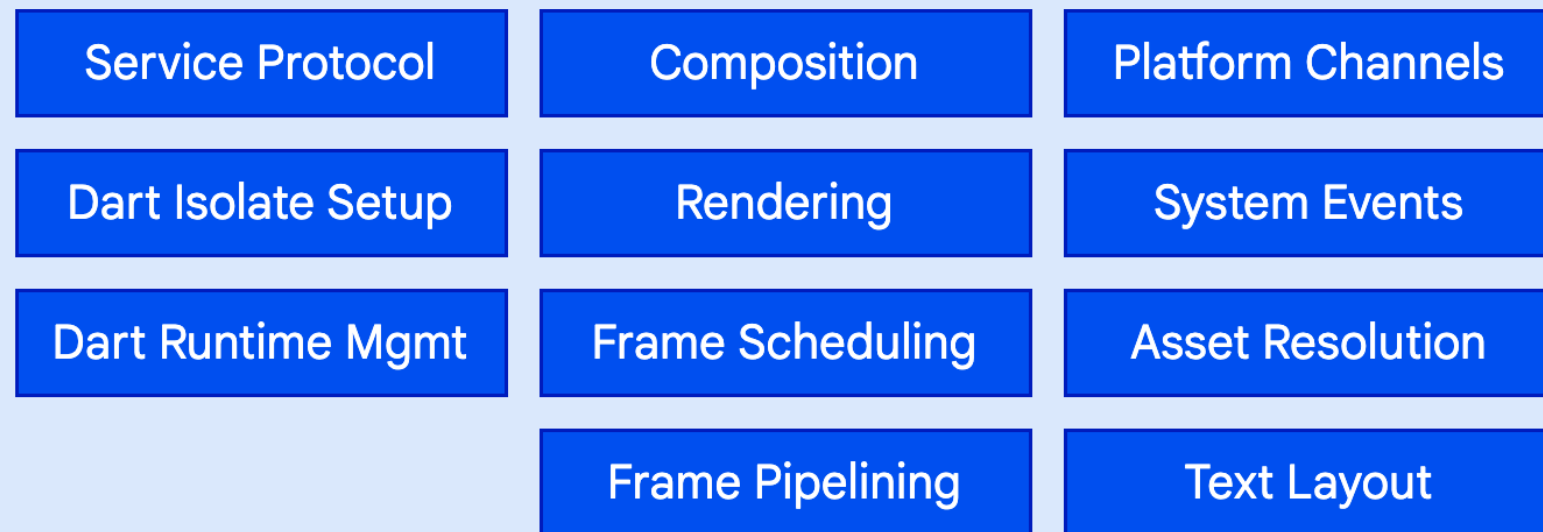
- Android Studio & IntelliJ



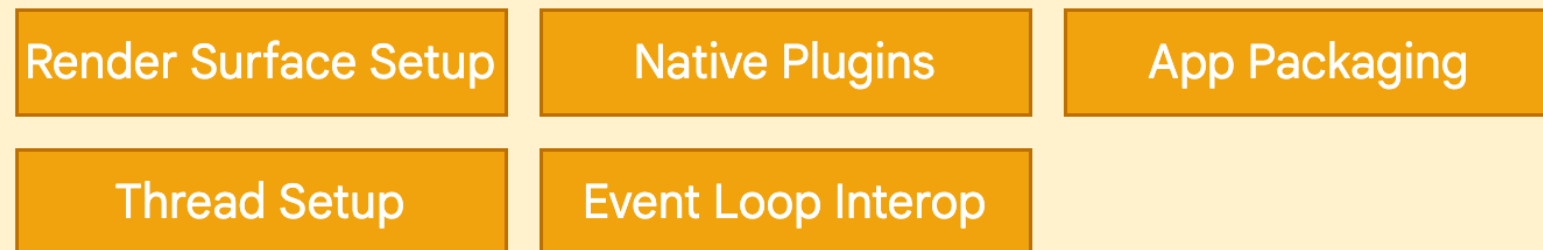
Framework
Dart

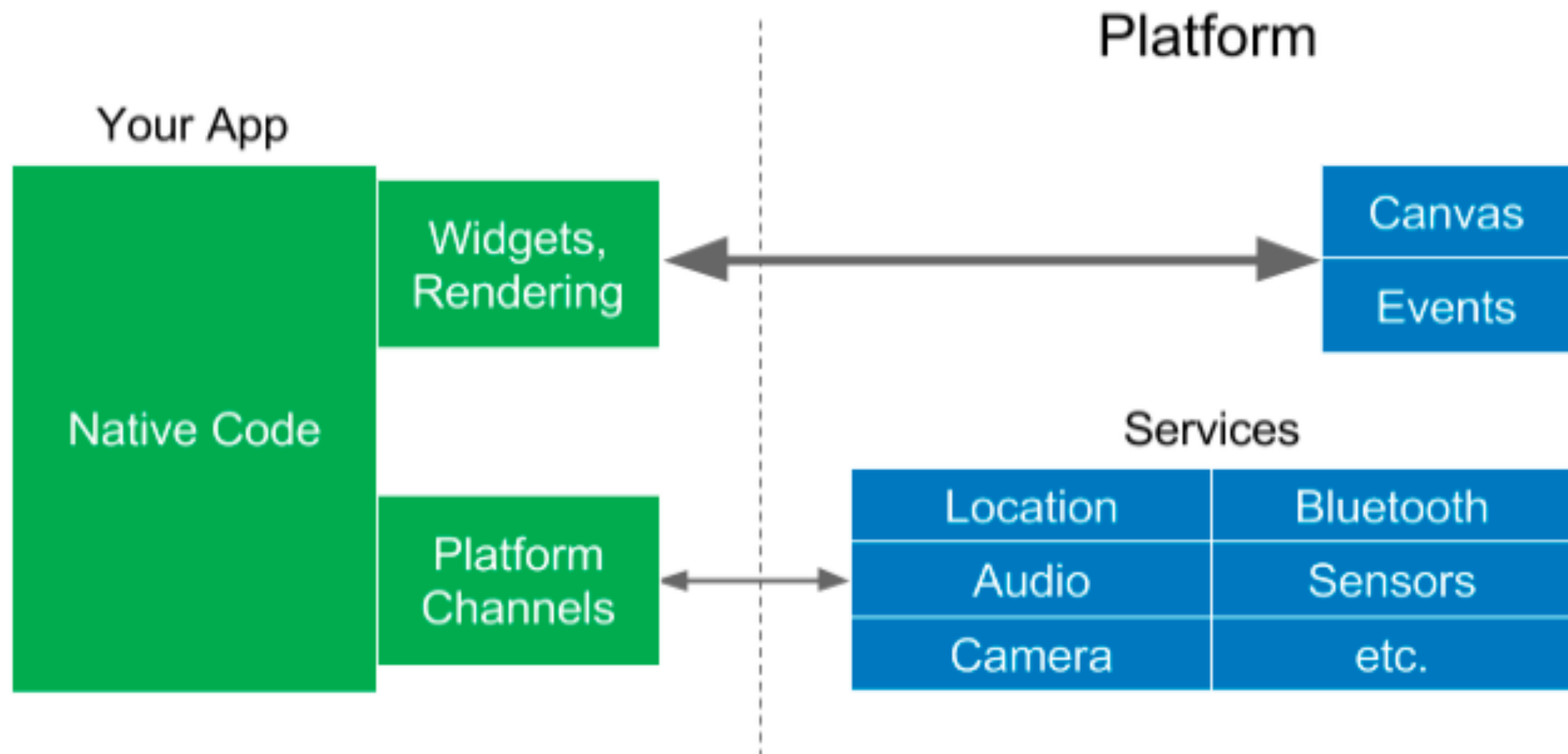


Engine
C/C++



Embedder
Platform-specific

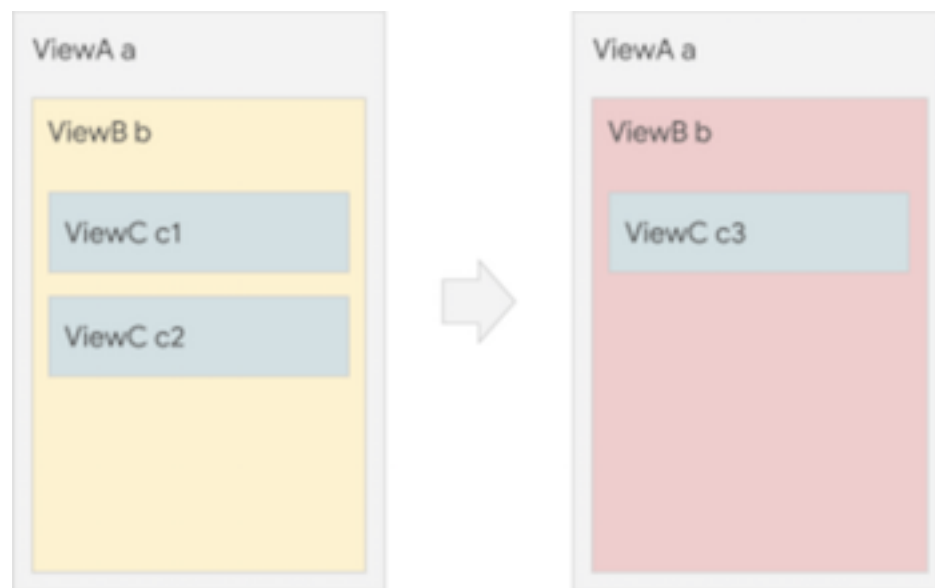




開發語言

Dart

- 物件導向語言
- 聲明式語法 Declarative style



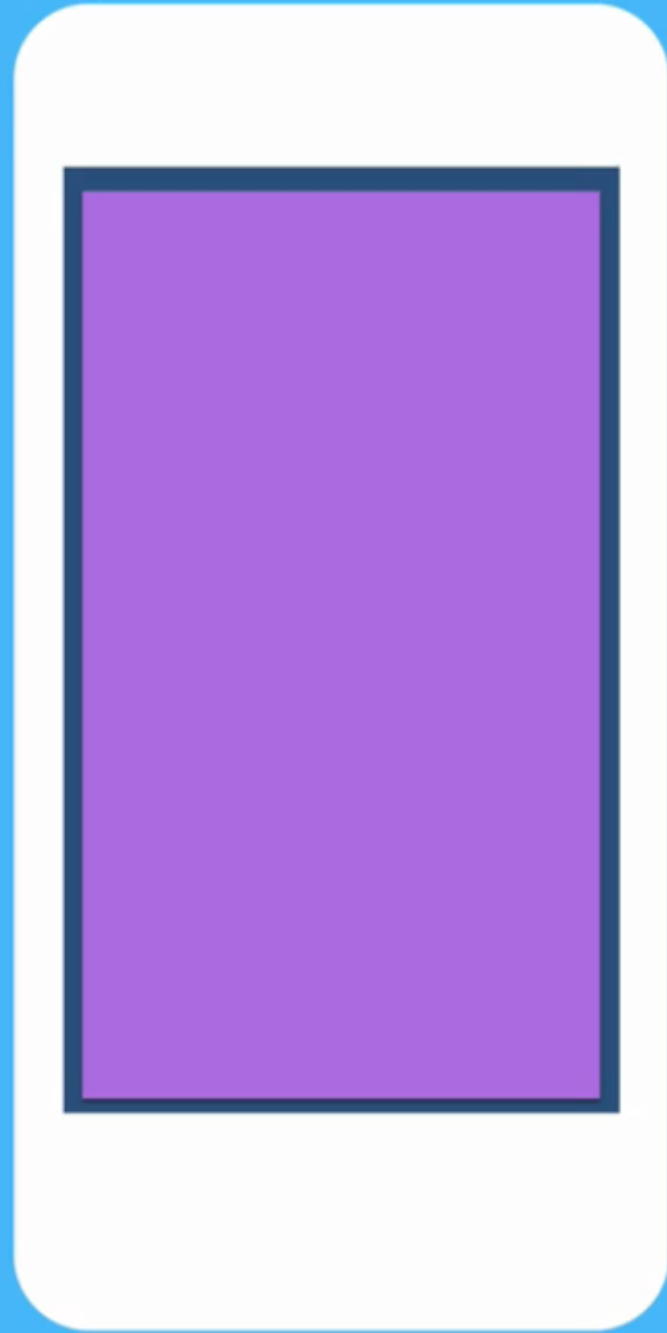
<https://flutter.dev/docs/get-started/flutter-for/declarative>

```
// Imperative style 命令式語法  
b.setColor(red)  
b.clearChildren()  
ViewC c3 = new ViewC(...)  
b.add(c3)
```

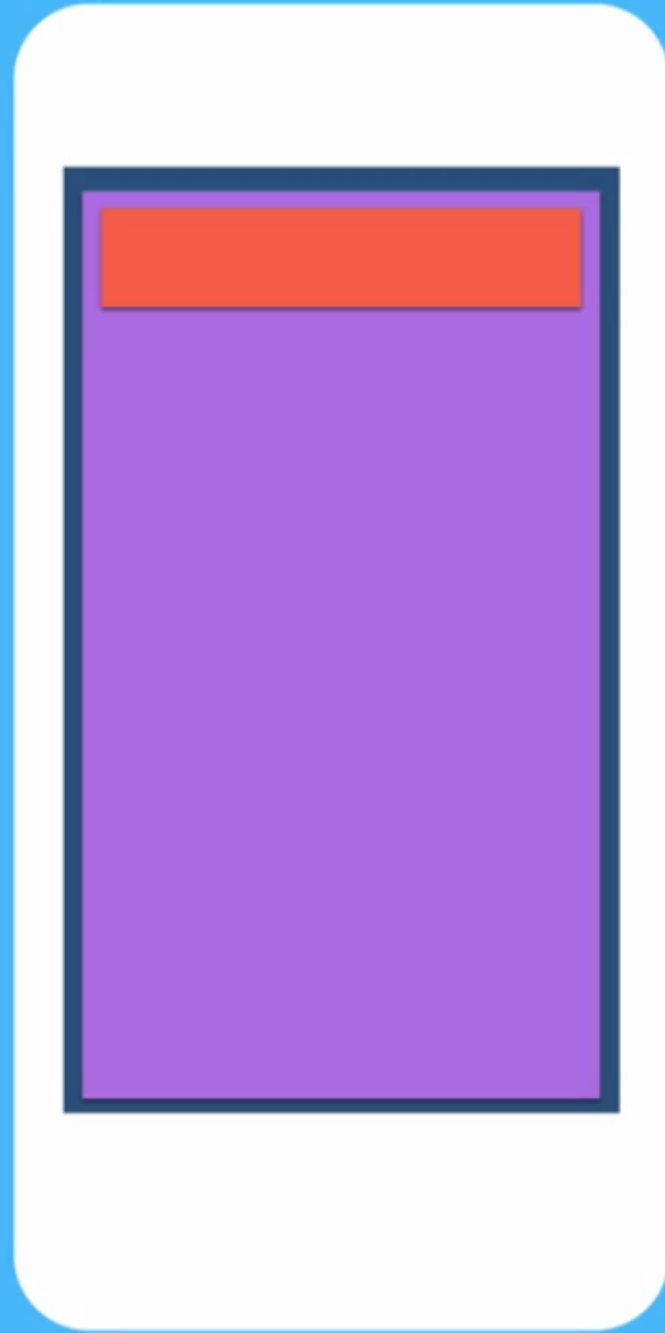
```
// Declarative style 聲明式語法  
return ViewB(  
  color: red,  
  child: ViewC(...),  
)
```

Everything's a Widget!

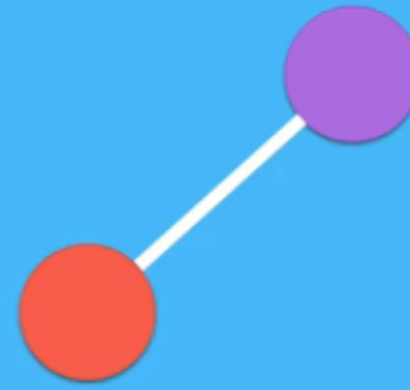




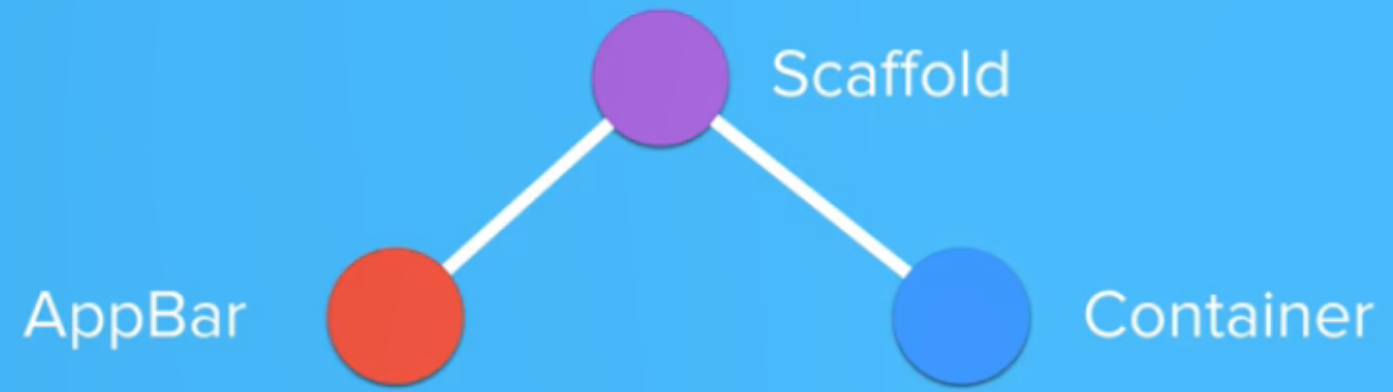
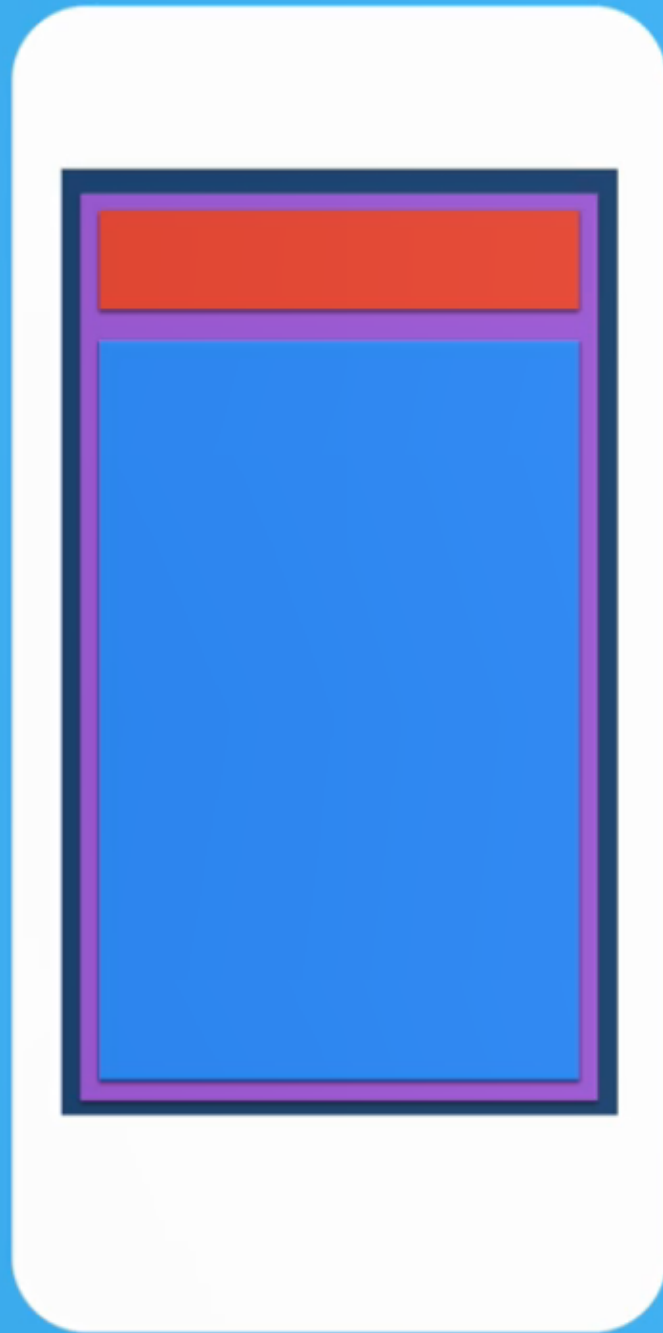
Scaffold

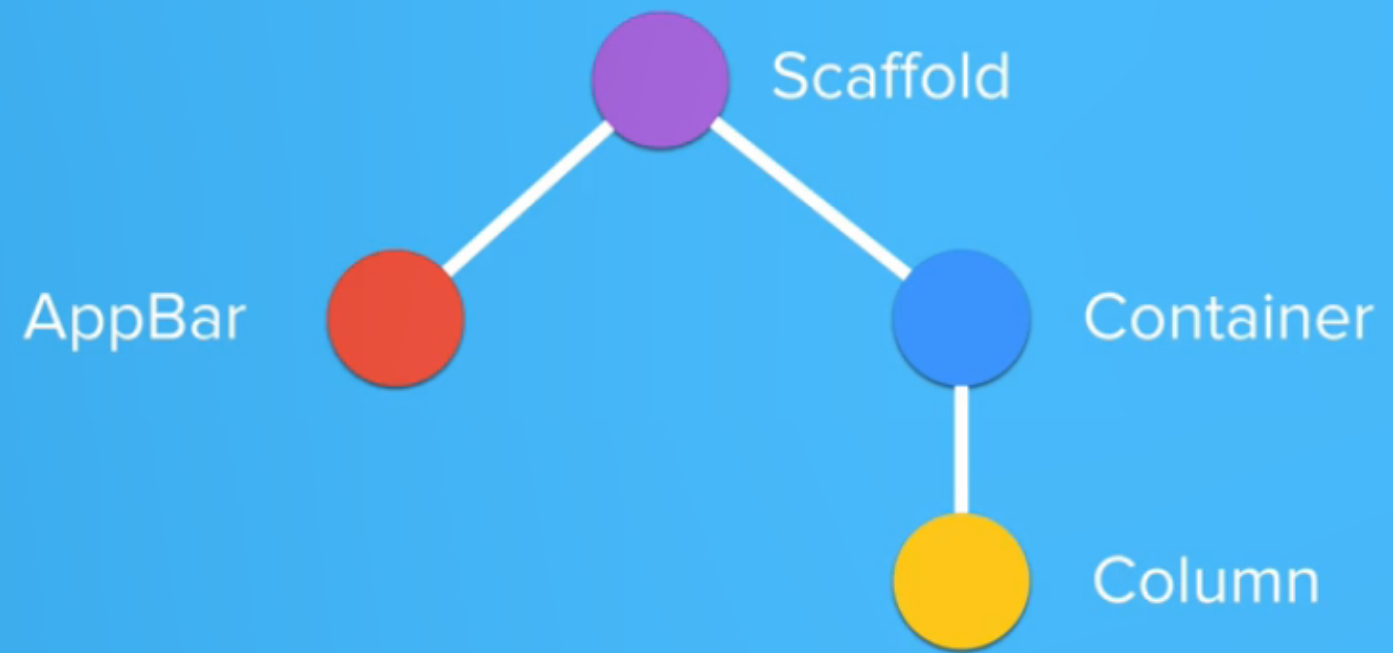
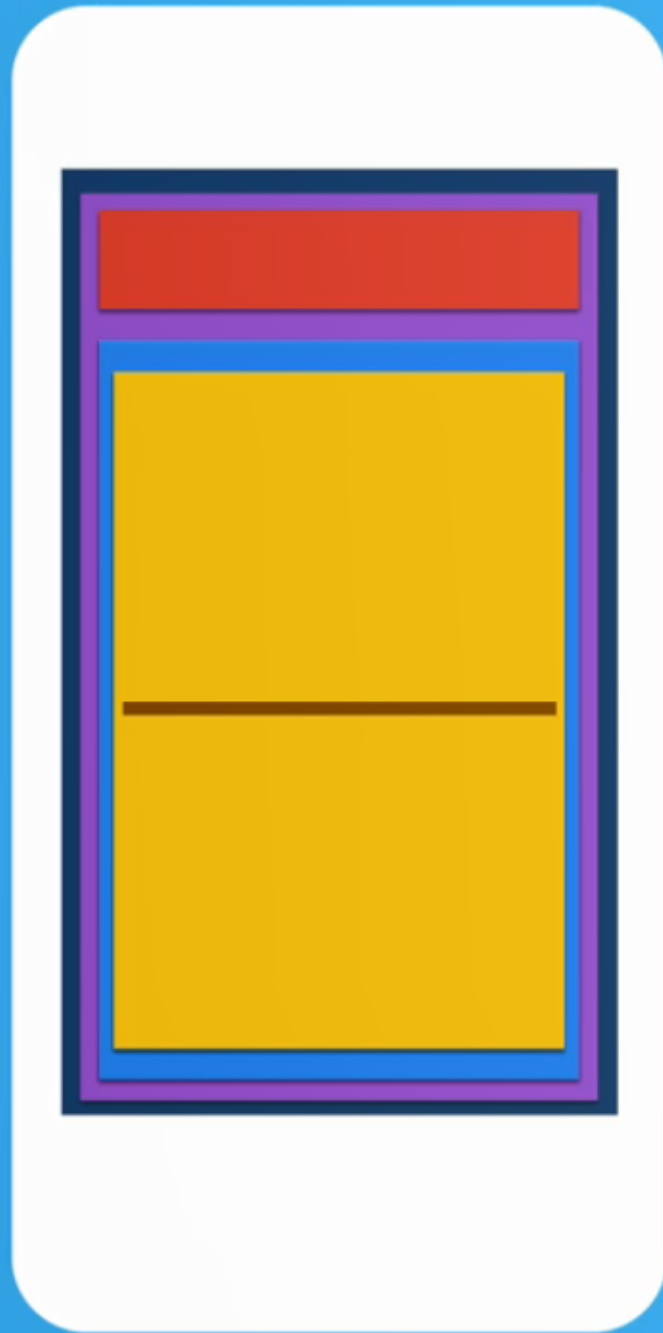


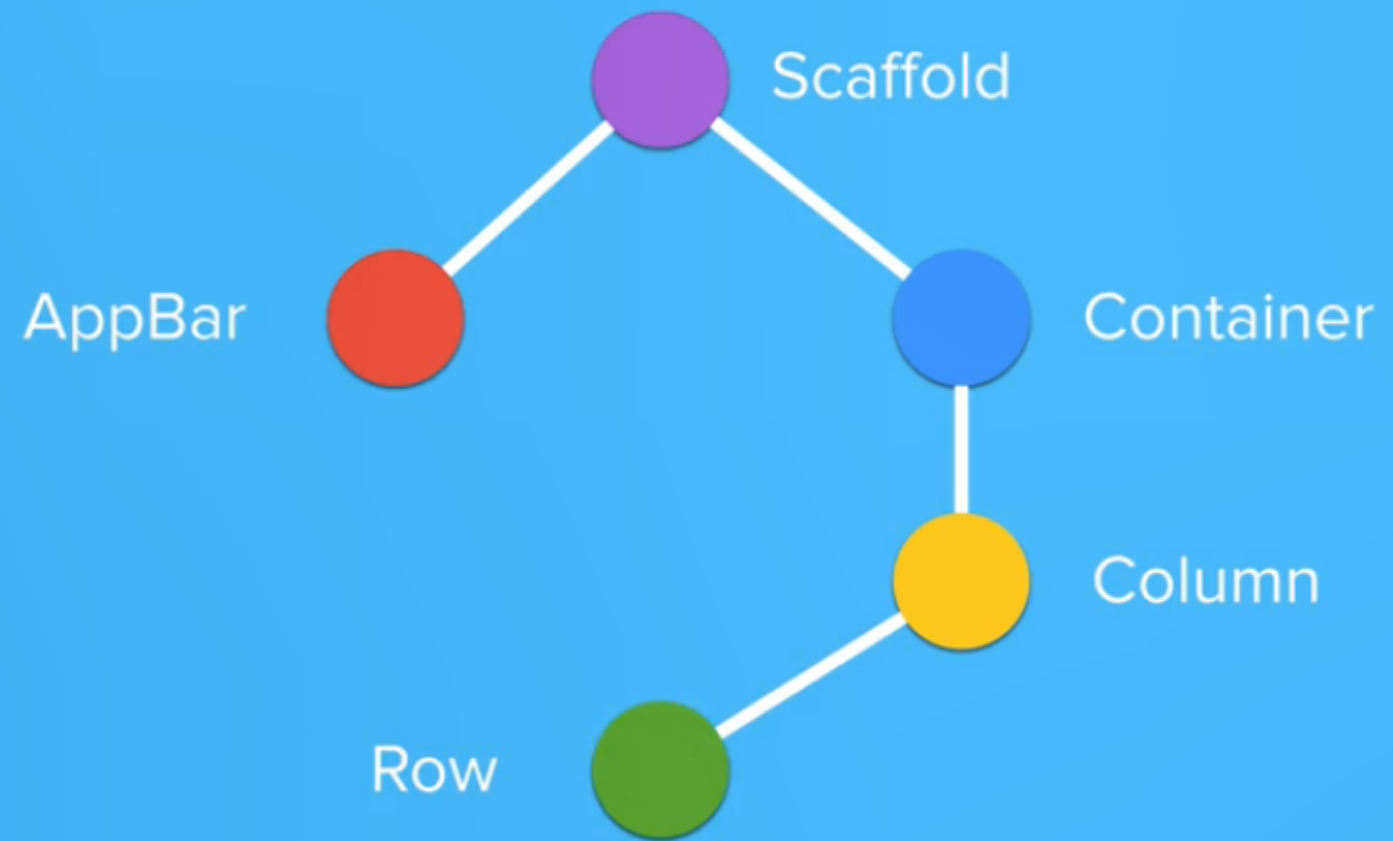
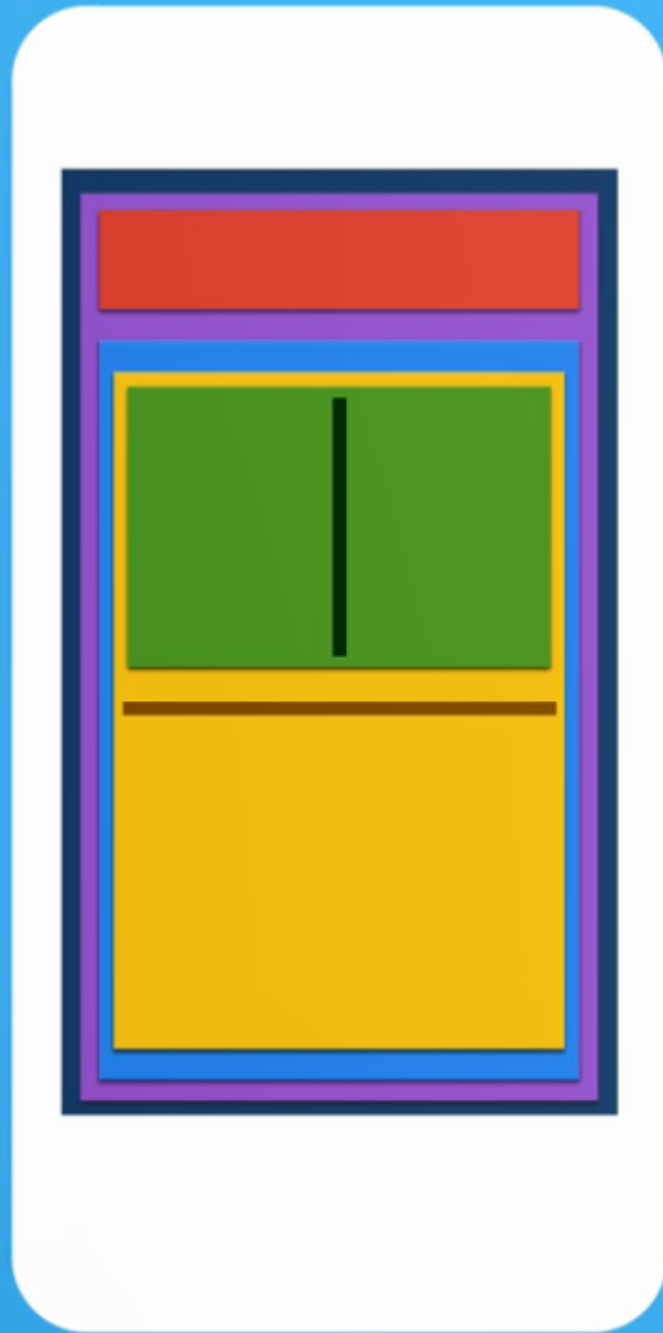
AppBar

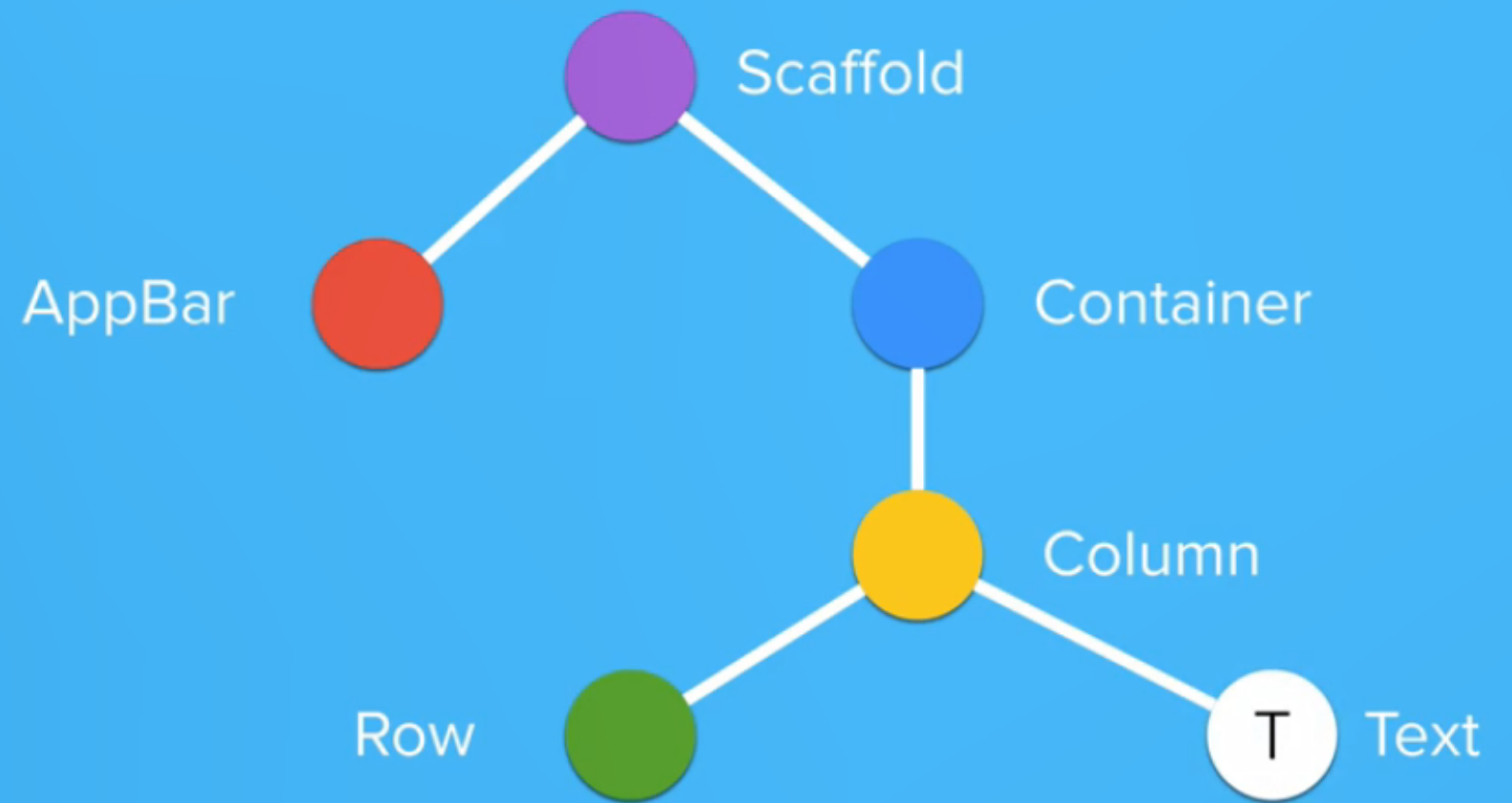
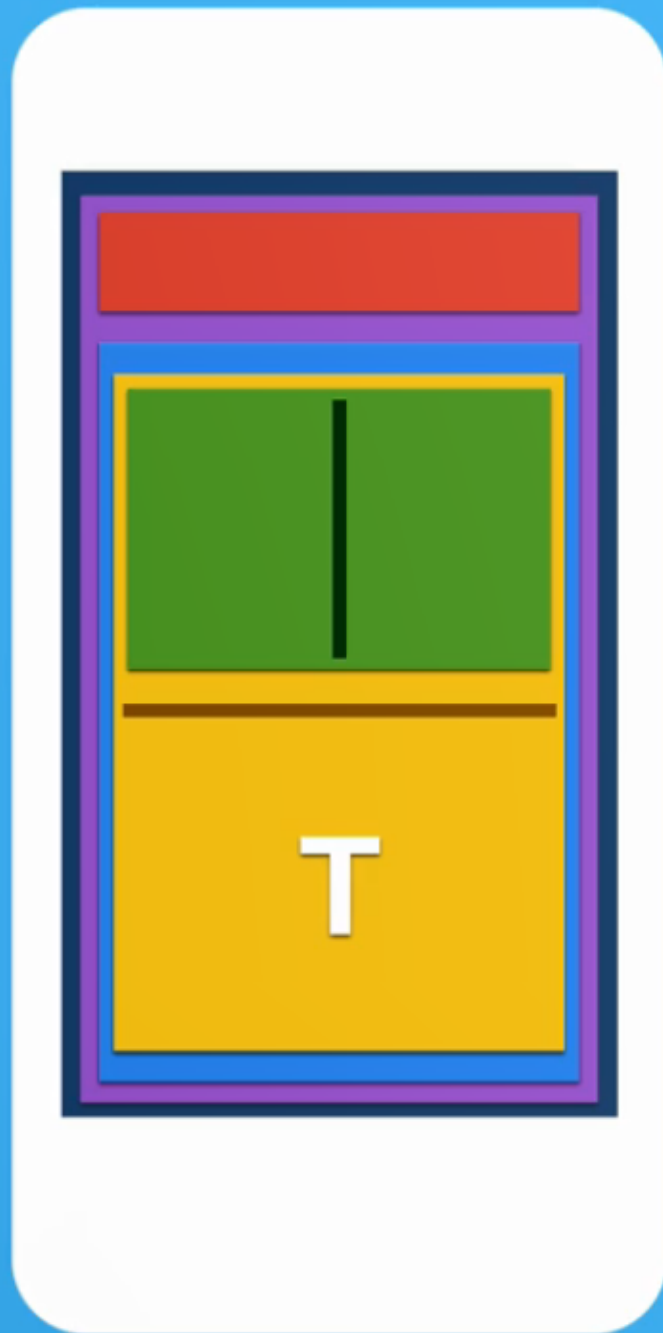


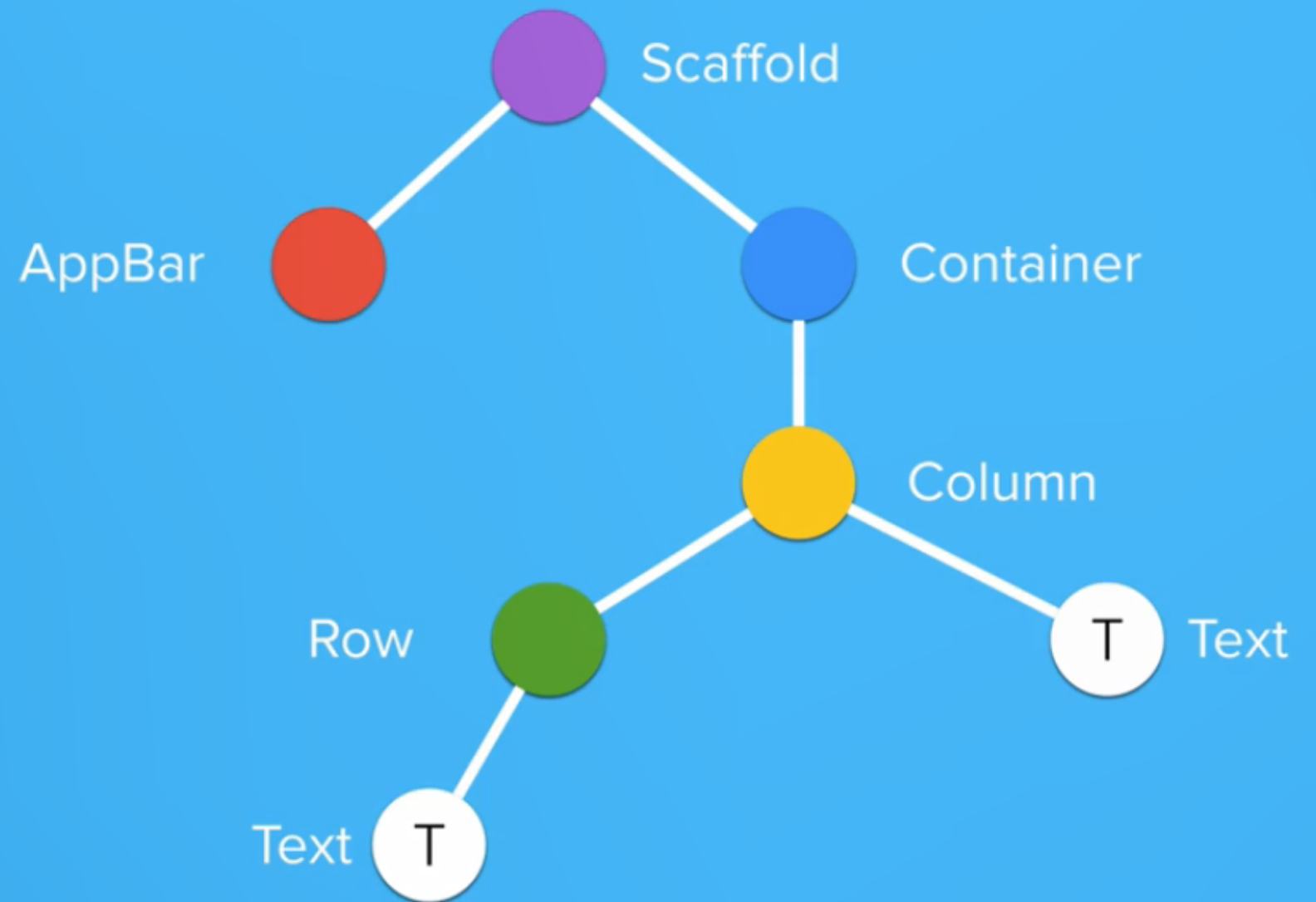
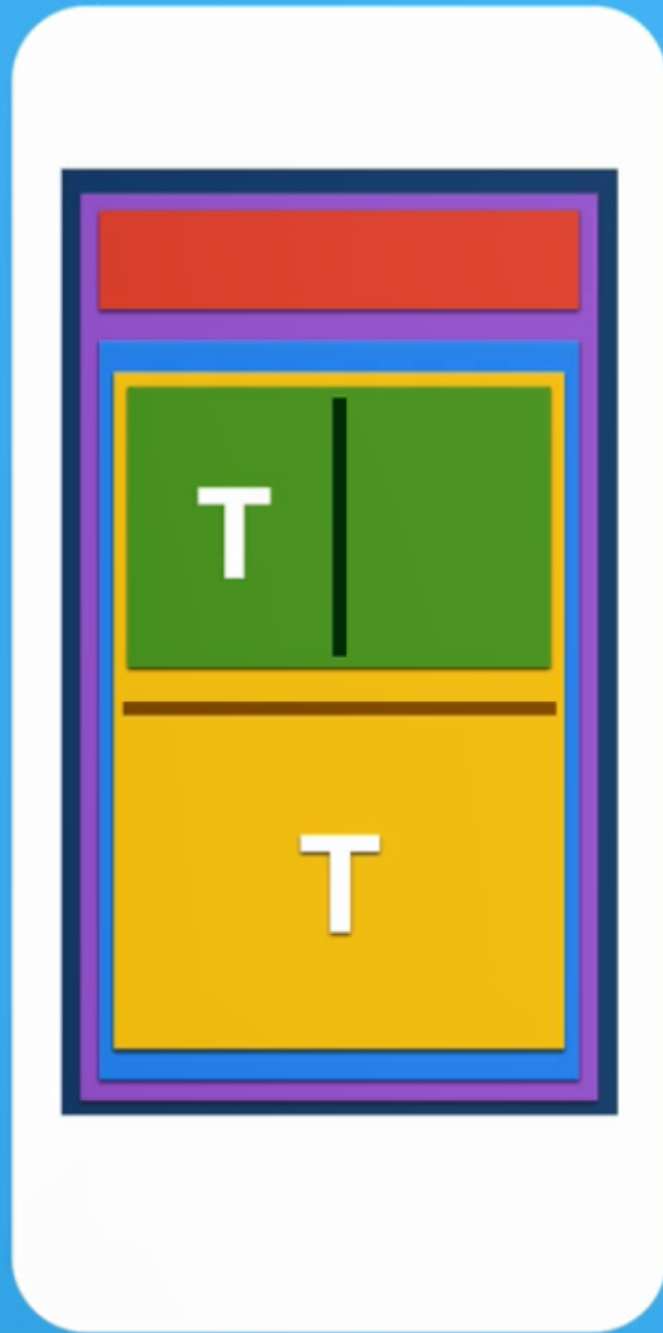
Scaffold

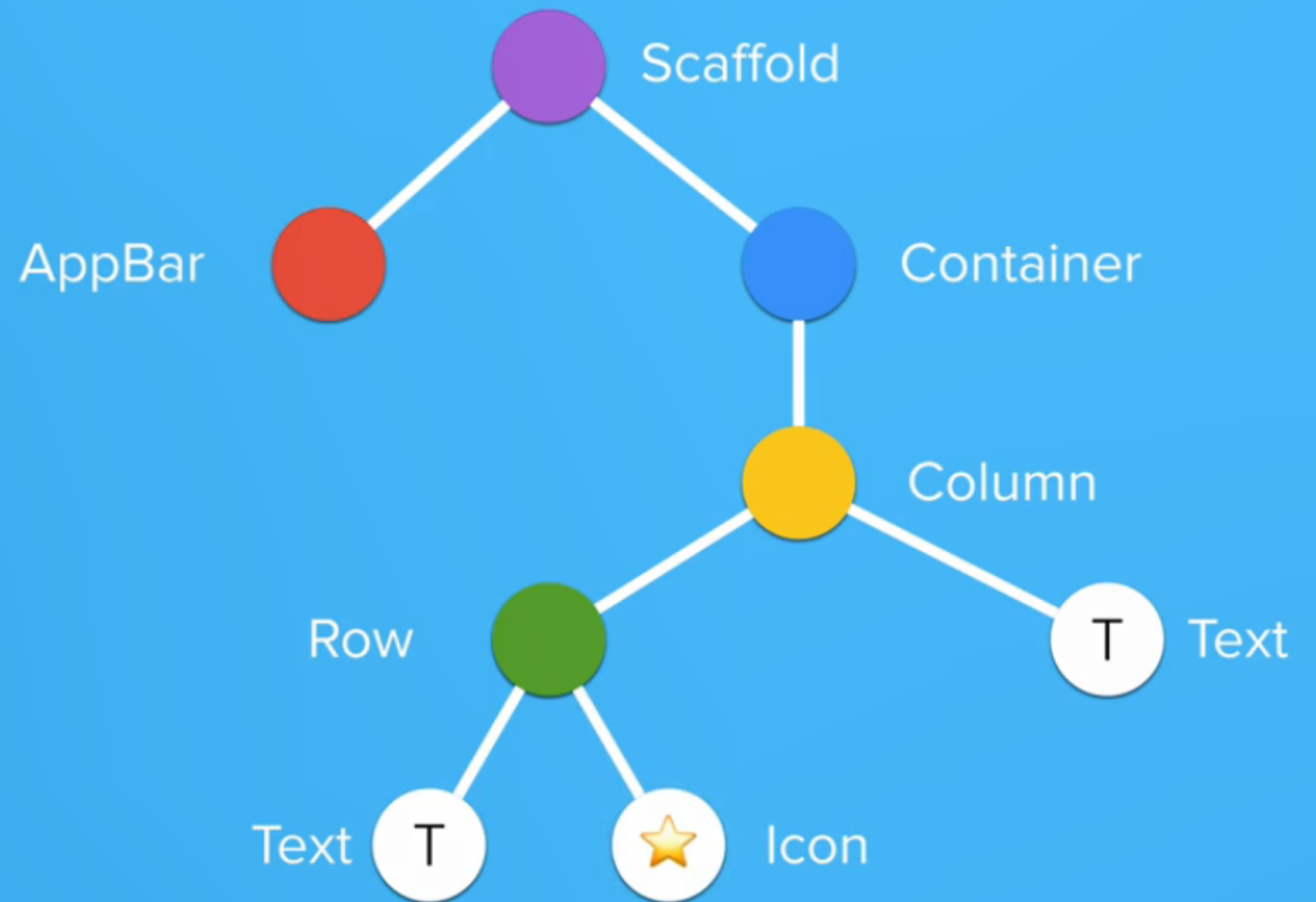
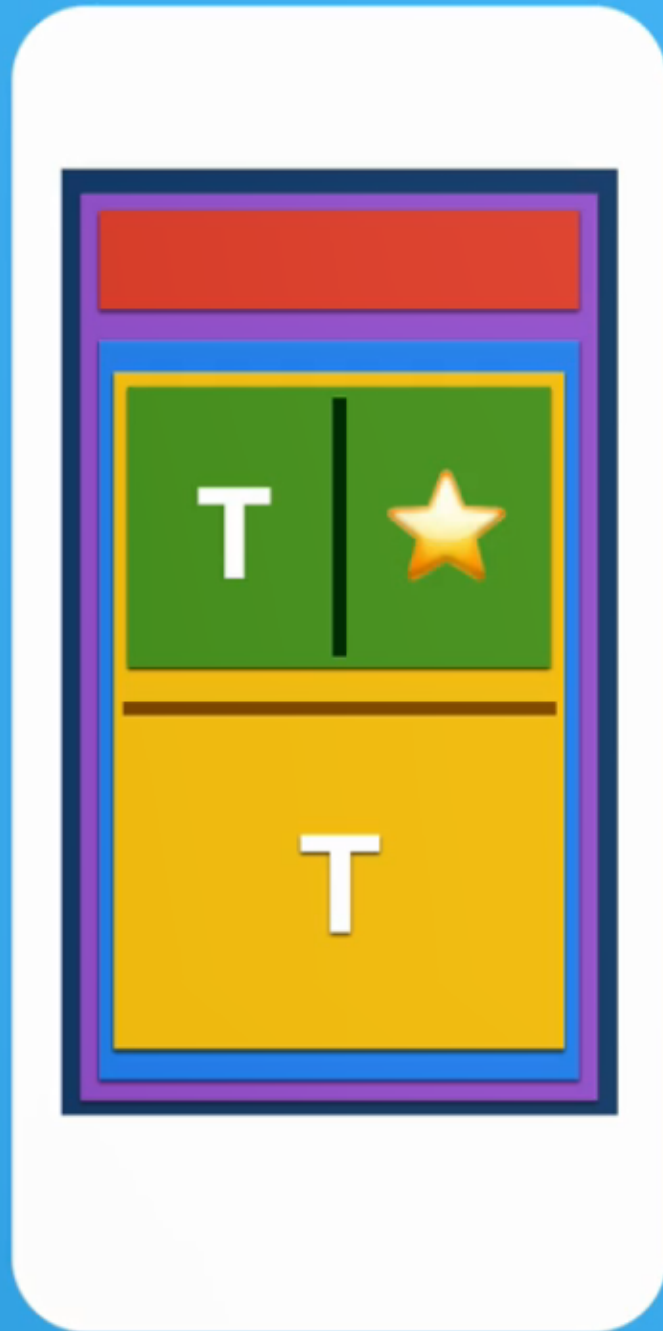


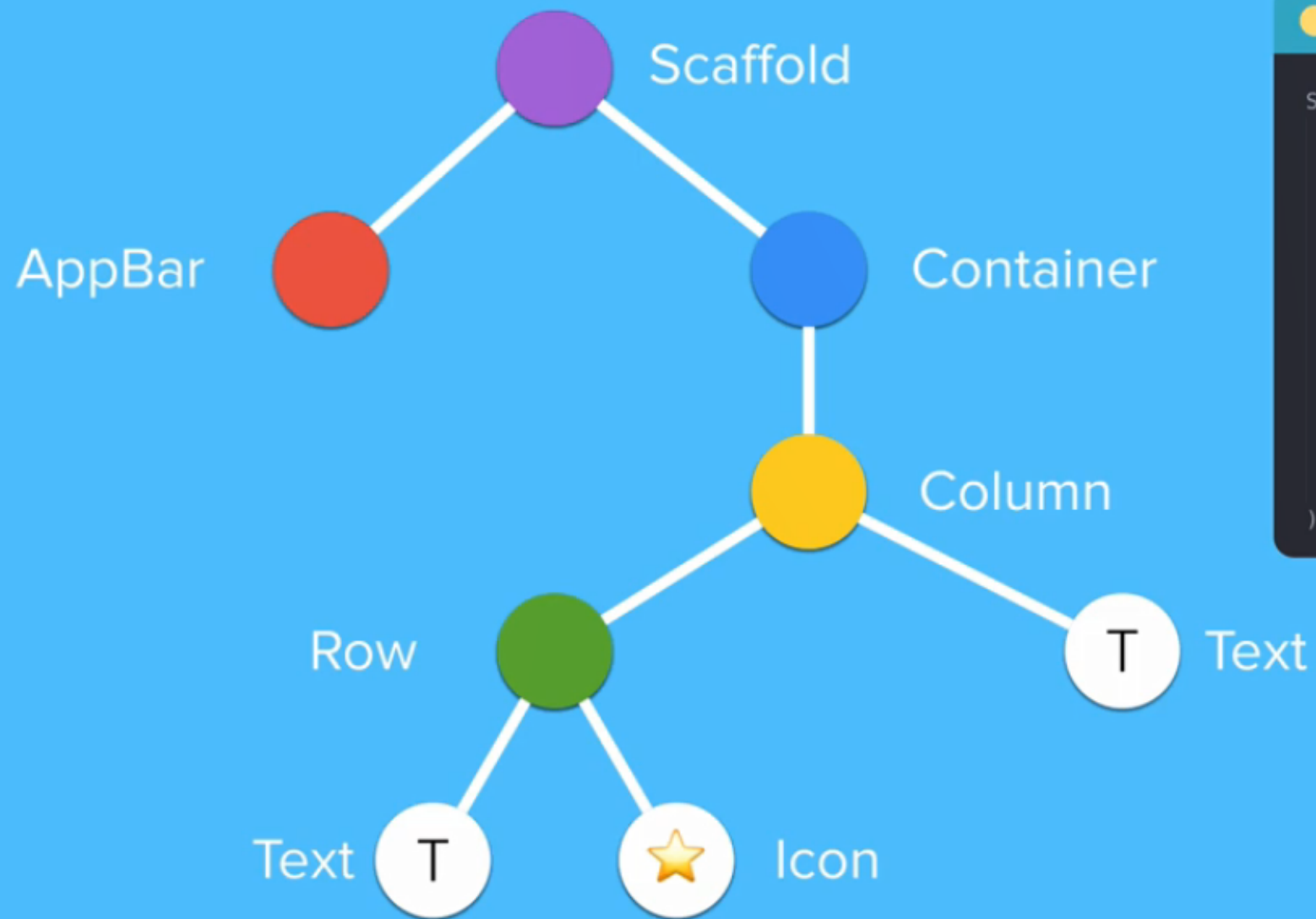












```
Scaffold(  
  appBar: AppBar(),  
  body: Container(  
    child: Column(  
      children: [  
        Row(  
          Text(),  
          Icon(),  
        ),  
        Text(),  
      ],  
    ),  
  ),  
)
```

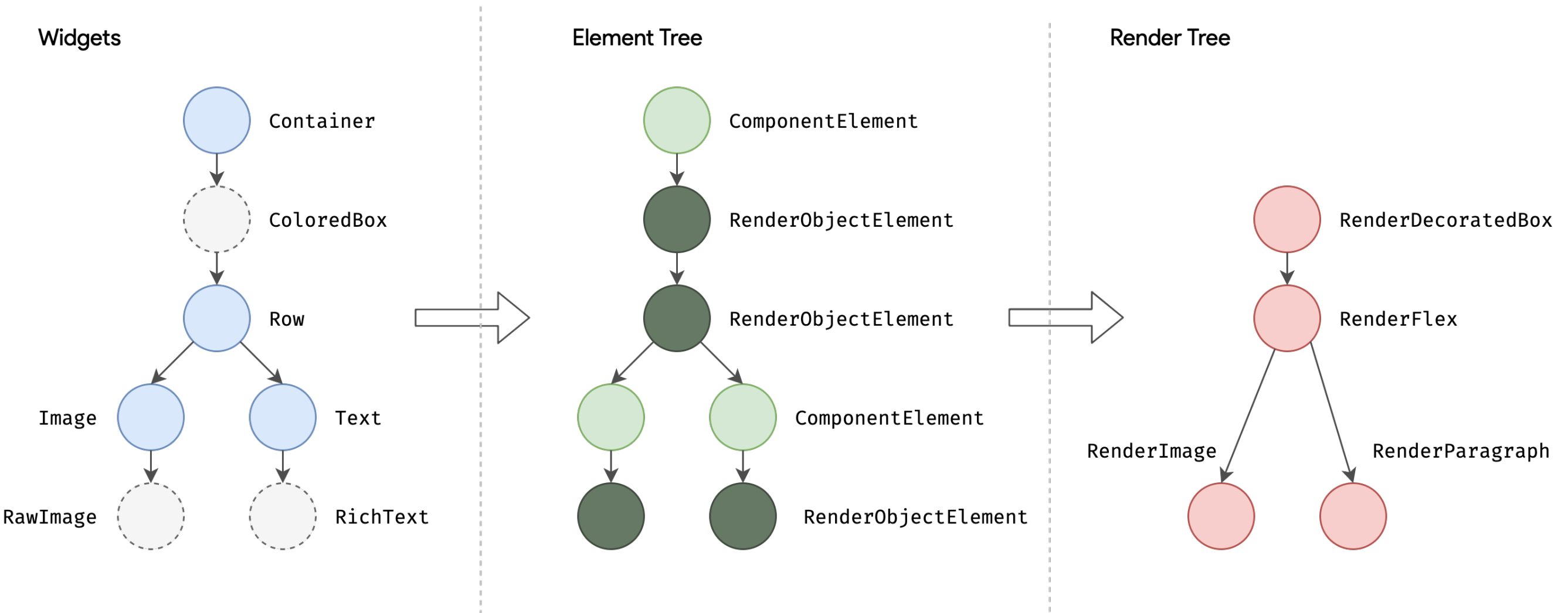
組件

Widgets

- Widget 是所有 Flutter 功能的抽象描述
- Tree 結構
- StatefulWidget v.s. StatelessWidget
 - UI 會不會受到外力影響而改變
- 若想修改 UI，那麼你需要 Rebuild 它
- <https://flutter.dev/docs/reference/widgets>

頁面渲染

Render Object



- Widget 是 element.config (藍圖的概念)
- Element Tree 是一一對應 Widgets 的實體物件，將最終真正需要繪製的部份轉化到 Render Tree

缺點

Cons

- 又要學新語言？！
- 現階段 Flutter 支援的 library 有限 (3DES...)
- 編譯出來的 App Size 會變大 (~5MB)
- 底層 Bug (WebView、鍵盤、Memory issue...)
- 複雜商業邏輯和資料變化，衍生出雜亂的狀態更新
- 恐怖的程式碼嵌套

安裝 Install

Get the Flutter SDK

1. Download the following installation bundle to get the latest stable release of the Flutter SDK:

`flutter_windows_1.22.5-stable.zip`

For other release channels, and older builds, see the [SDK releases](#) page.

2. Extract the zip file and place the contained `flutter` in the desired installation location for the Flutter SDK (for example, `C:\src\flutter`).

⚠ Warning: Do not install Flutter in a directory like `C:\Program Files\` that requires elevated privileges.

If you don't want to install a fixed version of the installation bundle, you can skip steps 1 and 2. Instead, get the source code from the [Flutter repo](#) on GitHub, and change branches or tags as needed. For example:

```
C:\src>git clone https://github.com/flutter/flutter.git -b stable
```



- <https://flutter.dev/docs/get-started/install>

CNOC 障礙通報 App

UI = **f(state)**

The layout
on the screen

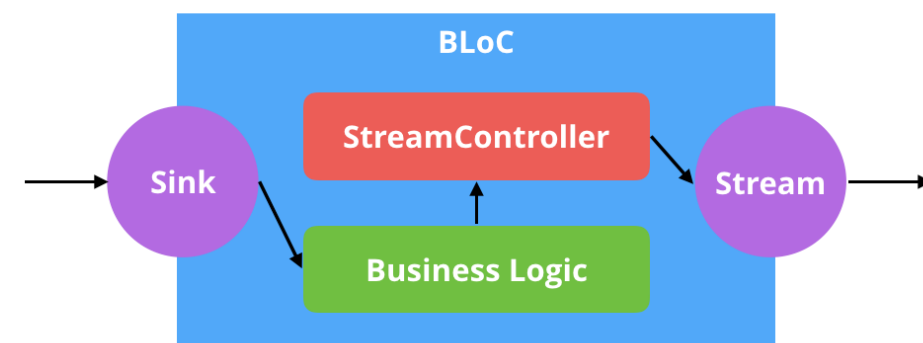
Your
build
methods

The application state

狀態管理

State Management

- BLoC
 - 登入頁、通報類型清單頁、
通報清單頁



- setState()
- MobX
- Provider

本地儲存

Local Storage

- Hive
 - No SQL
 - key-value
 - 用於儲存使用者資訊、API 快取
 - <https://pub.dev/packages/hive>

網路層

Network Layer

- `api_helper.dart`
 - `NetworkBaseHelper`
 - `ApiHelper`

模型

Model

- JSON 序列化/反序列化
 - json_serializable
 - Mapping: json_annotation
 - flutter packages pub get run build_runner build
- Dictionary
 - map["PARAM_CODE"]

基於 CNOC 的重用組件 Reusable Widget

- `utils.dart`
 - `cnocMultipleSelectWidget`
 - `cnocTextInputWidget`
 - `showAlertDialog`
 - ...

推播管理

PushNotificationsManager

- 串接 Firebase
- `push_notifications.dart`
- 取得 `fcmToken` 並儲存



資源管理

Resource Management

- pubspec.yaml
 - environment
 - dependencies
 - flutter pub get
- assets

佈署

Deployment

- Android
 - flutter build apk --release --split-per-abi --no-shrink --obfuscate --split-debug-info=obfuscate/symbols
 - 可直接包出 apk 提供測試
- iOS
 - flutter build ios --obfuscate --split-debug-info=obfuscate/symbols
 - 再於 Xcode - Runner Project(runner.xcworkspace) 執行 Archive , 最後才能發佈到 TestFlight
 - 需有 iOS 開發者帳號

What Else...

- 進階開發
 - Testing
 - CI/CD
 - State Management
 - 減少程式碼嵌套
 - Hot Keys
 - Stream
 - implements/extends/mixin
- Widget
 - 生命週期
 - 組合/代理/繪製
 - 佈局約束
 - setState() 原理
 - 對應 Element
 - navigator 2.0