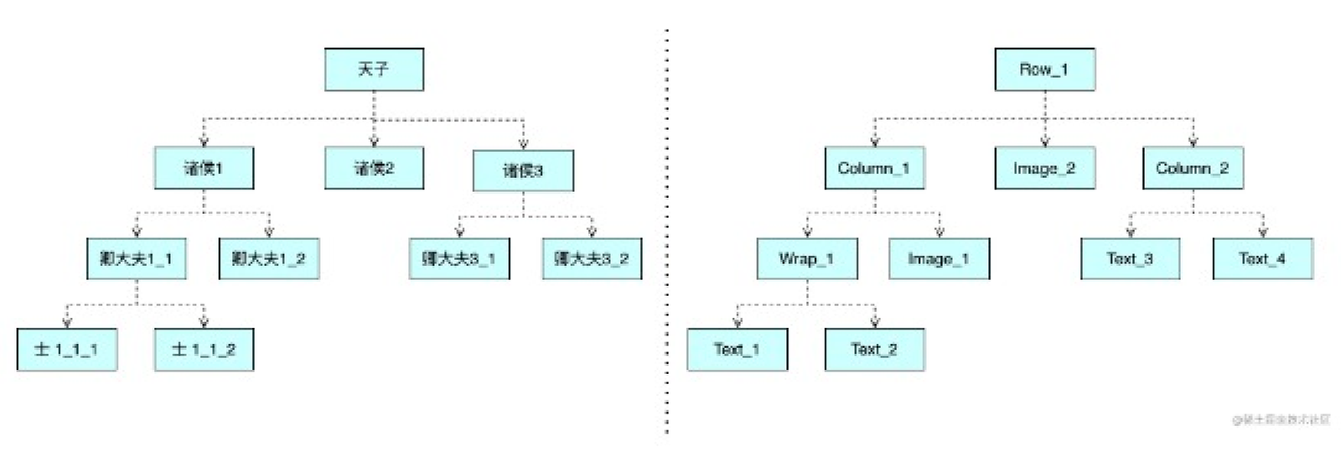
一、布局简述

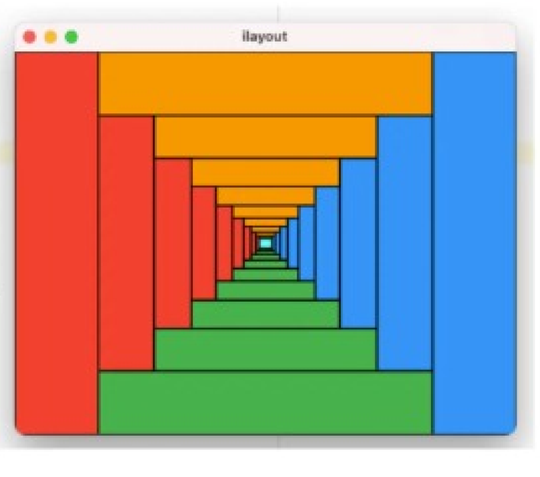
布局的重要性不言而喻，如果把 界面搭建 比作 盖房子 ，布局就是钢筋水泥，它是支撑界面呈现的 根基 。如果把 界面搭建 比作 征战沙场 ，布局就是 调兵遣将 ，呈现特定的阵型。

任何 框架， 任何 平台中的布局概念，都脱离不了两个字 嵌套 。可能说到嵌套，大家就会想到 嵌套地狱 。首先我们要认识到：对于布局来说，嵌套是不可避免的，嵌套是布局结构实现最便捷的方式；另外，嵌套会为布局提供一个非常重要的特性： 层级性 ，这个性质使其拥有 树状结构 。

这很像分封制，天子把土地分给各 诸侯 ，各诸侯再把土地分给各个 卿大夫 ，各个卿大夫再把土地分给 士 ，士分给 平民 。其中屏幕尺寸就像土地总区域，每个人获得土地后都可以向下一级分配。



只不过 分封制 是有限深度，而布局是无限可分封的。被分割出来的小块本身也可以看做一块新的布局区域，这样部分也可拥有整体的性质，就是 一花一世界 的体现，说专业点就是 自相似性 。



其实，这也就是 嵌套地狱 形成的根本原因。很多人初学者喜欢把所有的结构全部 写在一起 ，然后说 Flutter 布局体验太差。这就像一本书 不加标点符号 ，没有层次、没有结构一样，这样写文章并能怪罪别人看不懂。虽然实现了界面呈现，但是代码一塌糊涂。布局上 “一气呵成” 不是 Flutter 该有的样子，组件的 封装 和 提取 对于层次的划分非常重要，可以使结构更精简，也便于复用。

二、布局研究的重点

对于一块布局而言，最核心的有三个要点：

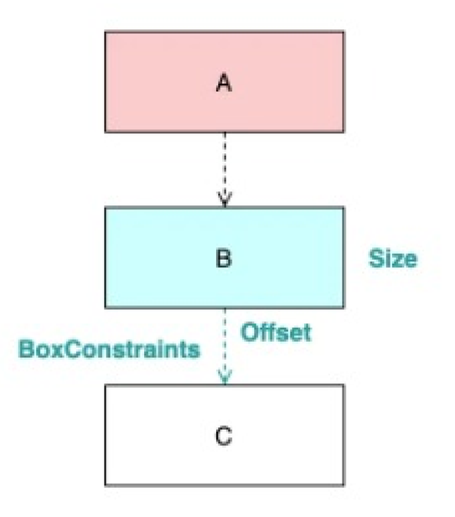
```css
1.我需要控制 [自身区域] 有多大 --- Size
2.我需要限制 [子级区域] 的范围 --- BoxConstraints
3.我需要确定 [子级区域] 偏移量 --- Offset
```

---

1. 明确目标

研究一个组件的布局特性，首先我们要确定研究目标所处的 场景 。就像人在不同场景会充当不同的角色，在父亲面前，自己是儿子；在儿子面前自己是父亲。对于有层级结构个体的研究，不明确目标很容易转晕。

比如下面是布局的层级结构， A 是 B 的父级， B 是 C 的父级，首先要明确研究谁的布局特性。比如确定研究 B 的布局特性时：需要重点关注的是 B 的尺寸是如何确定的； B 向 C 传递了什么约束； B 会让 C 产生多大的偏移量。如果想研究 A 向 B 传递了什么 BoxConstraints ，需要在你的意识里把研究目标切换到 A 。

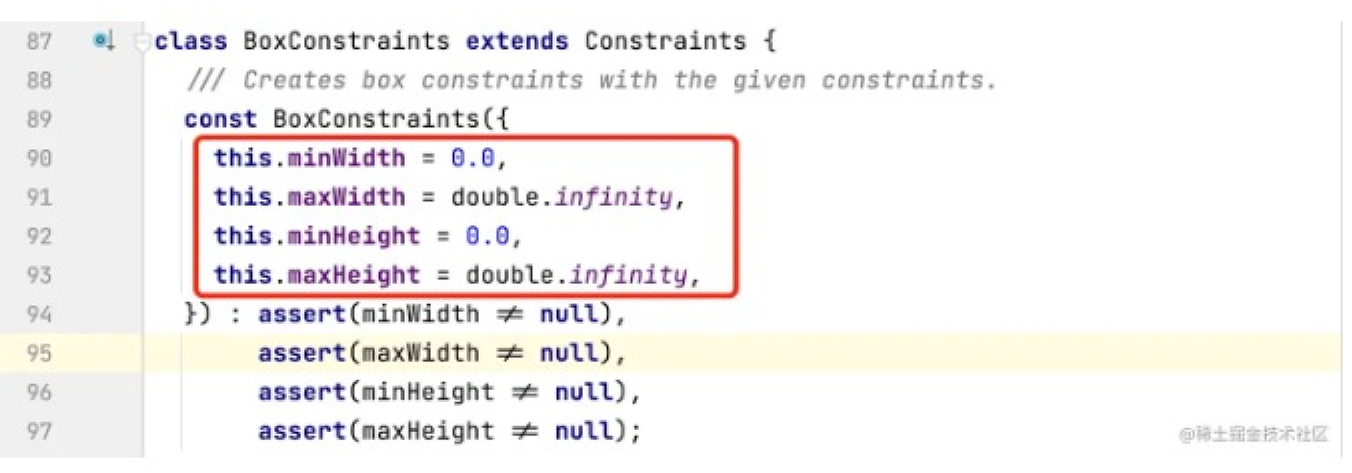


---

2. 什么是约束

约束是 Flutter 布局的 独有特性 ，也是对布局来说 最 为重要的概念。其实约束很好理解，就是 限制条件 。比如薪资范围 8~10K ，限制的是工资的上下界；年龄 24~30 岁 限制的是年龄上下界。所以我们要先明白 Flutter 的约束 限制的是什么 。

Flutter 中的约束通过 Constraints 类进行抽象，其中只有 BoxConstraints 和 SliverConstraints 两种实现类。 SliverConstraints 是在滑动中的约束，在 《Flutter 滑动探索 - 珠联璧合》 中做过详细介绍。非滑动组件中的约束指的都是盒约束 BoxConstraints ，也是本册研究的重心。

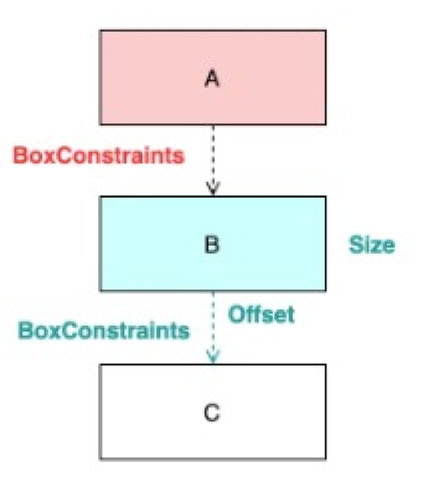


```dart
class BoxConstraints extends Constraints {
  /// Creates box constraints with the given constraints.
  const BoxConstraints({
    this.minWidth = 0.0,
    this.maxWidth = double.infinity,
    this.minHeight = 0.0,
    this.maxHeight = double.infinity,
  }) : assert(minWidth != null),
       assert(maxWidth != null),
       assert(minHeight != null),
       assert(maxHeight != null);
```

从上面 BoxConstraints 定义中可以看出，其本质就是维护 宽高 两个维度的 范围 。从后面章节对源码的分析可用知道， 父级传递约束 就是用来 确定子级尺寸 的，另外约束在布局中的一大特点是 传递性 ：

如下图所示：对于 B 而言，它所 受到的约束 是由 A 传递的红色约束。该约束会用来确定 B 的尺寸，并且不同组件确定尺寸的逻辑不同，这个逻辑就是其布局的 尺寸特性 。

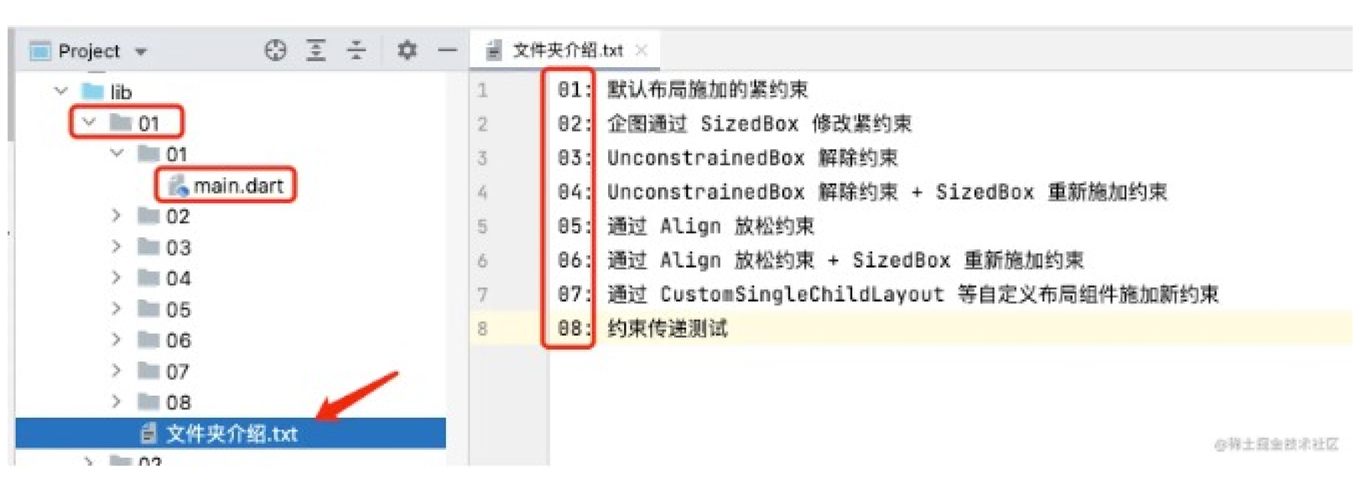
同理， B 会也会传给 C 一个约束，用于限制 C 的尺寸，并且不同组件确定传递约束的逻辑不同，这个逻辑就是其布局的 约束特性 。



尺寸 和 约束 是所有布局组件都会有的特性，至于偏移量，只有某些组件会有。比如 Align 、 Padding 等，可以让子级相对于区域左上角产生偏移。 这里先对这些概念有个简单的认知即可，这些在后面的源码分析中会做详细解释。

三、本册源码的使用

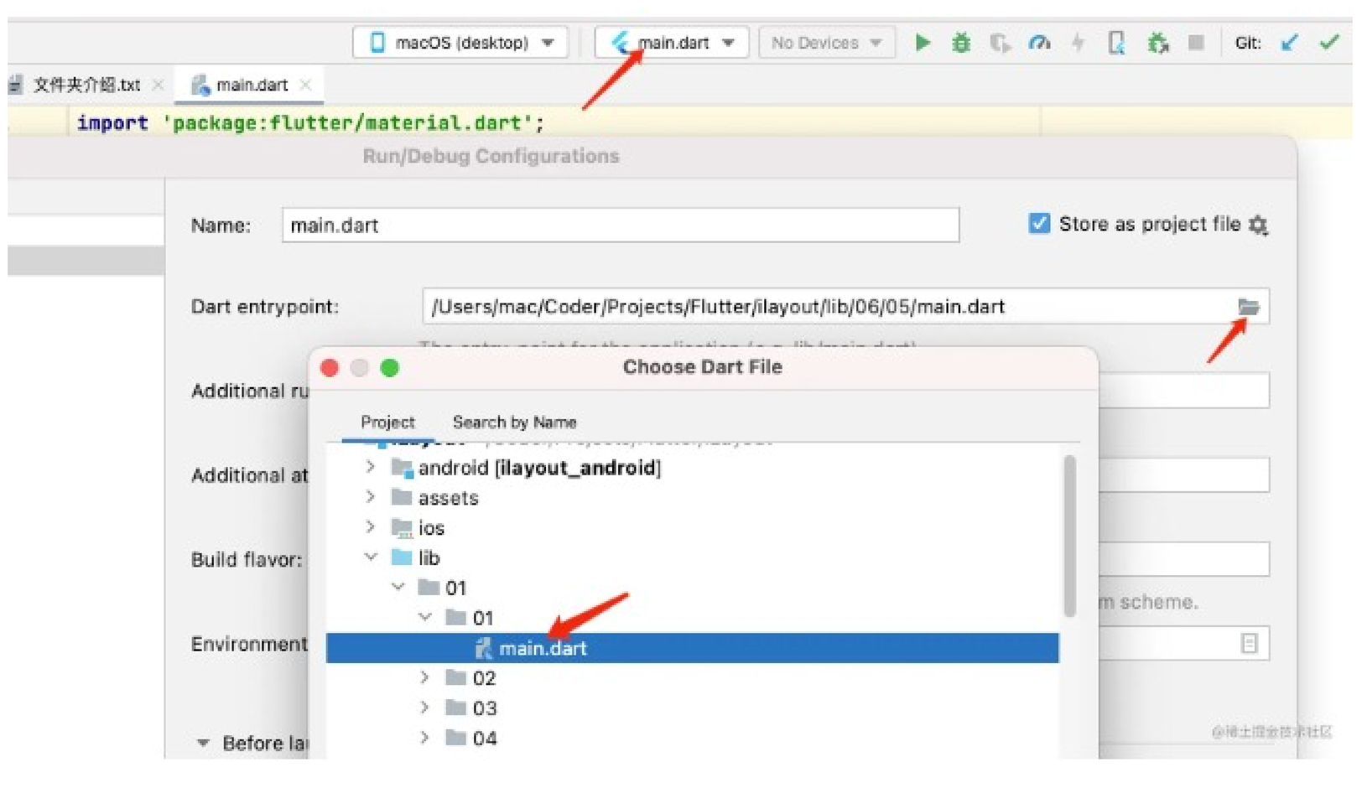
本册的所有源码在 【ilayout】 。每个章节对应一个数字文件夹，每个章节对应的测试案例也在数字文件夹中，其中每个测试案例都有一个 main.dart 的入口文件。另外每个章节中有一个 文件夹介绍 的文件，会说明每个数字文件夹的用途。



另外，正文中的代码上方也有相关标识，可以为你快速定位到源码位置，相信通过这样，你可以更好地食用小册。

```dart
---->[01/01]----
void main() {
  runApp(const ColoredBox(color: Colors.blue));
}
```

在 AndroidStudio 中，可以通过如下方式来修改运行文件的入口，这样可以更快速地切换测试案例。



最后，本册中有大量源码的分析，当前版本为 Flutter 2.10.2 。

```dart
Flutter 2.10.2 • channel stable • https://github.com/flutter/flutter.git
Framework • revision 097d3313d8 (4 weeks ago) • 2022-02-18 19:33:08 -0600
Engine • revision a83ed0e5e3
Tools • Dart 2.16.1 • DevTools 2.9.2
```

那废话不多说，就让我们一起开始 Flutter 布局的探索之旅吧 ~

留言

输入评论（Enter换行，⌘ + Enter发送）

发表评论

全部评论（15）

juedui0769 JY.4 | 3月前
控制、限制、确定，有点强行加概念，不如本来意思：大小、约束、偏移；大佬的课程都买了（支持大佬😄），这里只是单纯发表下看法...
点赞 1

> 张风捷特烈（作者） | 3月前
> 无所谓吧，那块只是一个口语化的表达。没必要上升到 [概念]😊
> 1 回复

衿璃 JY.4 | flutter | 4月前
动画太难了换本简单的看看🙂
1 回复

了不起的王小一 LV.1 JY.4 | 5月前
写的挺好的，调理性很好，也很深入，为什么出一个完整版的?
点赞 回复

用户3397150600... JY.1 | 7月前
开始学习
点赞 回复

Jjiawen LV.1 JY.5 | 资深CV工程师 | 7月前
目前flutter 3.0 和 flutter 2.0 区别还是有不少的，这个会升级么?
点赞 2

> 张风捷特烈（作者） | 7月前
> 这些基础的东西，区别应该不大。demo 有空我都升一下
> 点赞 回复
>
> Mr_羊 回复 张风捷特烈 | 5月前
> 现在都是3.x了，会更新吗
> “这些基础的东西，区别应该不大。demo 有空我都升一下”
> 点赞 回复

咸鱼不咸618 JY.3 | 8月前
打卡
点赞 回复

juedui0769 JY.4 | 9月前
github链接打不开
点赞 回复

比巴抱 JY.3 | 10月前
打开
点赞 回复

新小梦 LV.5 JY.5 | Android @ 快速记账 | 11月前
如果想研究 A 像 B 传递了什么 BoxConstraints。有错别字
点赞 1

> 张风捷特烈（作者） | 11月前
> get
> 点赞 回复

biandong LV.1 JY.3 | Golang全栈 @ 他念他翁 | 11月前
占位
点赞 回复

不会写代码的iOSer LV.3 JY.4 | iOS开发工程师 | 11月前
前排
点赞 回复

格外关注 LV.1 JY.4 | BFE | 11月前
打卡!
点赞 回复