

Flutter 扩展NestedScrollView (三) 下拉刷新的解决



法的空间 LV.6

2018年11月14日 22:27 · 阅读 8757

[Extended NestedScrollView](#) 相关文章

- [Flutter 扩展NestedScrollView \(一\)Pinned头引起的bug解决](#)
- [Flutter 扩展NestedScrollView \(二\)列表滚动同步解决](#)
- [Flutter 扩展NestedScrollView \(三\)下拉刷新的解决](#)

现在推荐大家食用[Flutter 下拉刷新花式玩法](#),支持各种下拉刷新姿势。

解决点之前的2个问题。开心的上项目试了下。完美。

FlutterCandies QQ群:181398081 [pub v6.0.0](#)

但是在使用官方的下拉刷新 [RefreshIndicator](#) 发现没法使用。

默默打开了源码,我们再来看一下。

首先,我调试到这个,发现notification.depth不为0,其实也好理解,因为NestedScrollView里面有很多能滚动的东西。默认的RefreshIndicator要求的是必须是第一层的它才其效果。

```
/// A [ScrollNotificationPredicate] that checks whether
/// `notification.depth == 0`, which means that the notification did not bubble
/// through any intervening scrolling widgets.
bool defaultScrollNotificationPredicate(ScrollNotification notification) {
  return notification.depth == 0;
}
```

arduino 复制代码

那么我改成,再试试呢?

```
bool defaultScrollNotificationPredicate(ScrollNotification notification) {
  return true;
  return notification.depth == 0;
}
```

arduino 复制代码

在_handleScrollNotification方法中,我们可以看到会有很多ScrollNotification进来,不同的,当你滑动在一个不能滚动的list里面的时候,获取的viewportDimension是为0.。这会覆盖掉之前有viewportDimension的值。

所以我做了以下改动

```
double maxContainerExtent = 0.0;
bool _handleScrollNotification(ScrollNotification notification) {
  if (!widget.notificationPredicate(notification)) return false;
  maxContainerExtent = math.max(
    notification.metrics.viewportDimension, this.maxContainerExtent);
  if (notification is ScrollStartNotification &&
    notification.metrics.extentBefore == 0.0 &&
    _mode == null &&
    _start(notification.metrics.axisDirection)) {
    setState(() {
      _mode = _RefreshIndicatorMode.drag;
    });
  };
  return false;
}
bool indicatorAtTopNow;
switch (notification.metrics.axisDirection) {
  case AxisDirection.down:
    indicatorAtTopNow = true;
    break;
  case AxisDirection.up:
    indicatorAtTopNow = false;
    break;
  case AxisDirection.left:
  case AxisDirection.right:
    indicatorAtTopNow = null;
    break;
}
if (indicatorAtTopNow != _isIndicatorAtTop) {
  if (_mode == _RefreshIndicatorMode.drag ||
    _mode == _RefreshIndicatorMode.armed)
    _dismiss(_RefreshIndicatorMode.canceled);
} else if (notification is ScrollUpdateNotification) {
```

dart 复制代码

```

    if (_mode == _RefreshIndicatorMode.drag ||
        _mode == _RefreshIndicatorMode.armed) {
        if (notification.metrics.extentBefore > 0.0) {
            _dismiss(_RefreshIndicatorMode.canceled);
        } else {
            _dragOffset -= notification.scrollDelta;
            _checkDragOffset(maxContainerExtent);
        }
    }
}
if (_mode == _RefreshIndicatorMode.armed &&
    notification.dragDetails == null) {
    // On iOS start the refresh when the Scrollable bounces back from the
    // overscroll (ScrollNotification indicating this don't have dragDetails
    // because the scroll activity is not directly triggered by a drag).
    _show();
}
} else if (notification is OverscrollNotification) {
    if (_mode == _RefreshIndicatorMode.drag ||
        _mode == _RefreshIndicatorMode.armed) {
        _dragOffset -= notification.overscroll / 2.0;
        _checkDragOffset(maxContainerExtent);
    }
} else if (notification is ScrollEndNotification) {
    switch (_mode) {
        case _RefreshIndicatorMode.armed:
            _show();
            break;
        case _RefreshIndicatorMode.drag:
            _dismiss(_RefreshIndicatorMode.canceled);
            break;
        default:
            // do nothing
            break;
    }
}
return false;
}
}

```

对于NestedScrollView 来说。我们只需要关注最大能滚动viewportDimension, 用这个来驱动整个下拉刷新。

Sample Code

用法跟官方一致

```

return NestedScrollViewRefreshIndicator(
  onRefresh: onRefresh,
  child: extended.NestedScrollView(

```

php 复制代码

最后放上 [Github extended_nested_scroll_view](#), 如果你有更好的方式解决这个问题或者有什么不明白的地方, 都请告诉我, 由衷感谢。

pub v6.0.0



