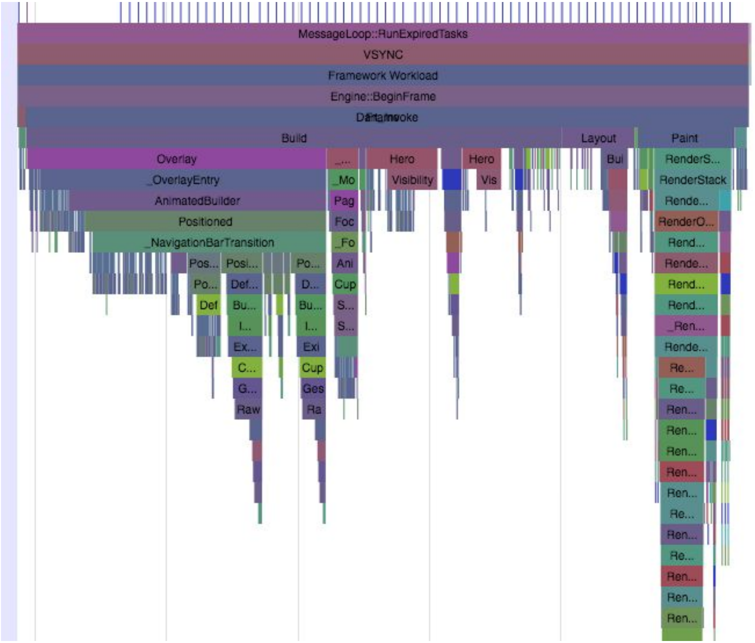
Flutter Performance Profiling and Theory
Flutter的性能测试与理论

Xiao Yu
于潇

听众收益

Talk Goals

- 时间都去哪了?
- 理念投入使用
- 使用工具调试



前言

Preface

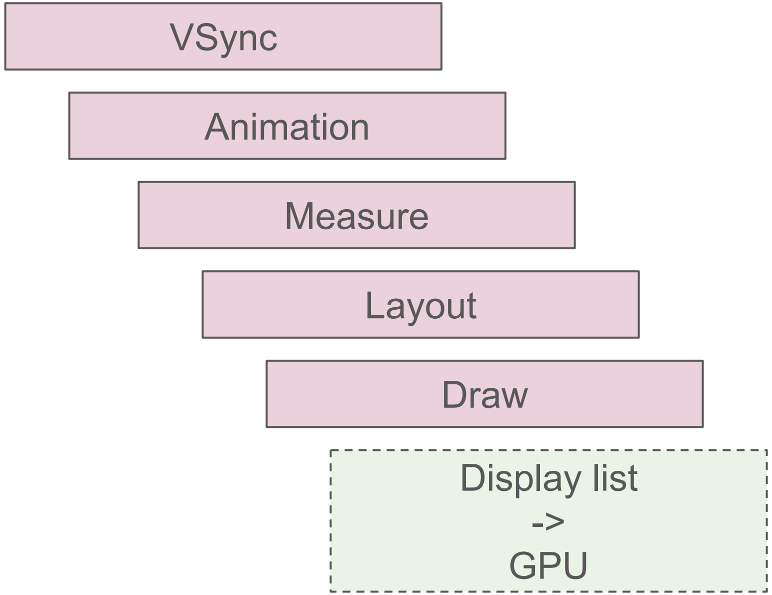
- 不是入门题材
- 同场5:30请关注Flutter图像系统性能论坛

什么是应用性能

What is Rendering Performance

什么是应用性能

What is Rendering Performance



VSync

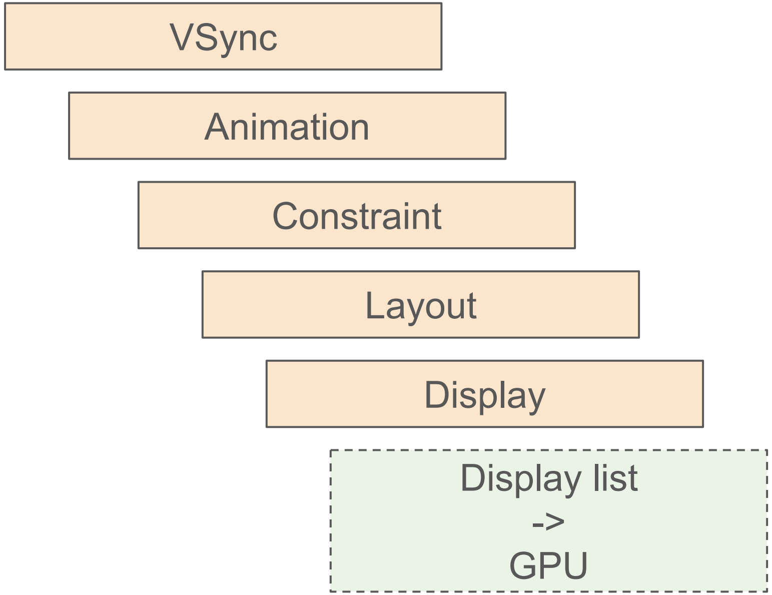
Animation

Measure

Layout

Draw

Display list
->
GPU



VSync

Animation

Constraint

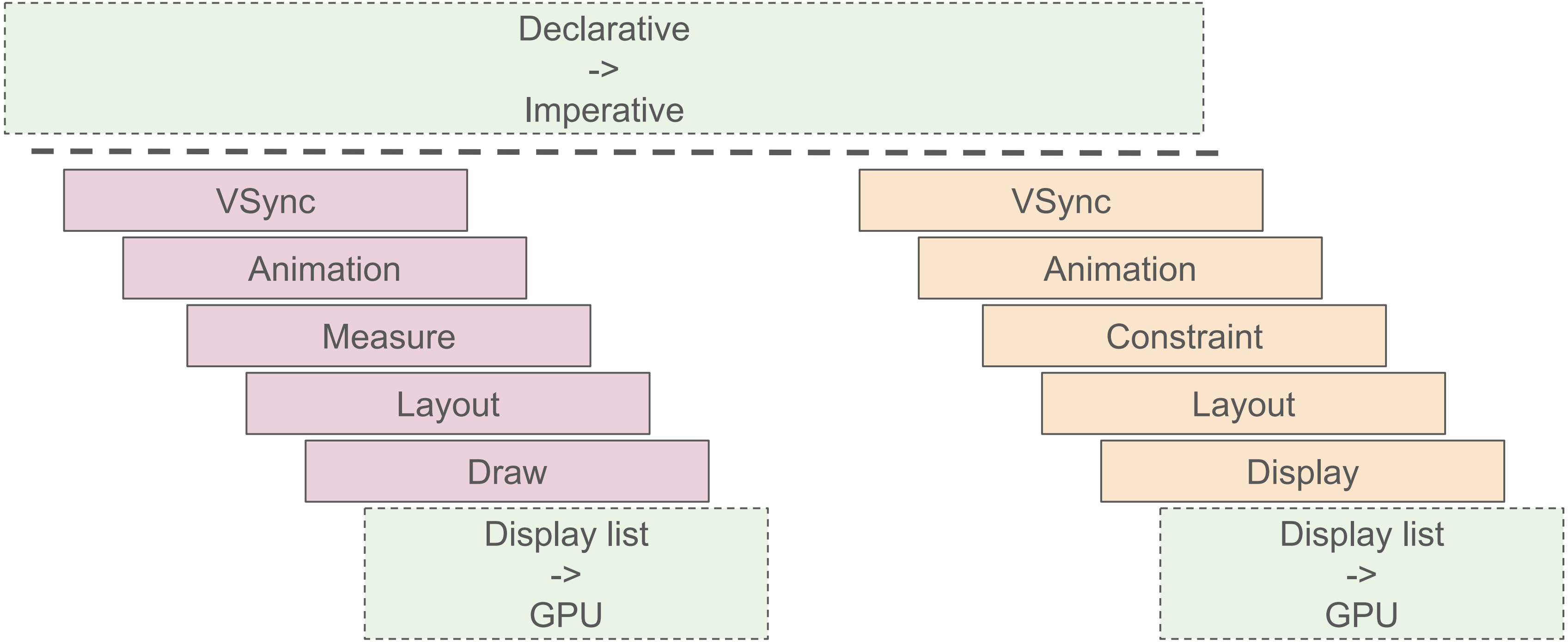
Layout

Display

Display list
->
GPU

什么是应用性能

What is Rendering Performance



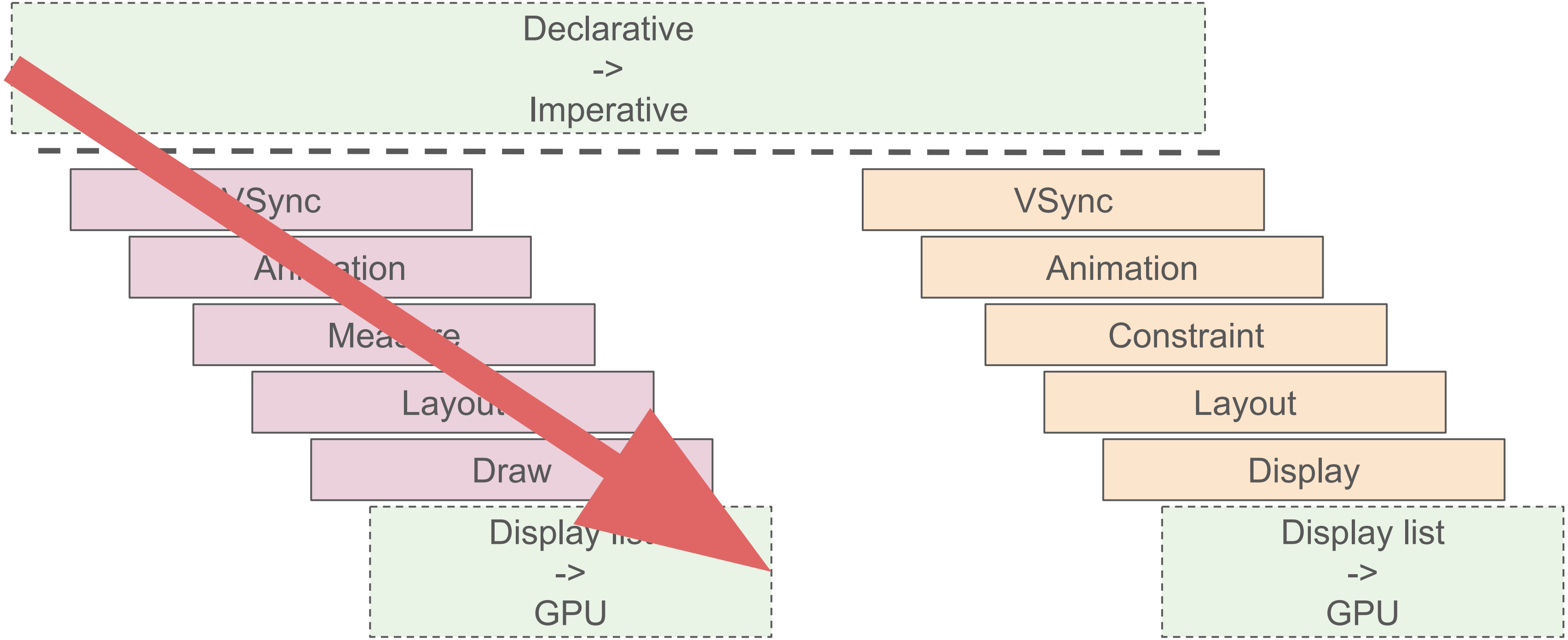
Declarative
->
Imperative

VSync
Animation
Measure
Layout
Draw
Display list
->
GPU

VSync
Animation
Constraint
Layout
Display
Display list
->
GPU

什么是应用性能

What is Rendering Performance



Declarative
->
Imperative

VSync

Animation

Measure

Layout

Draw

Display list
->
GPU

VSync

Animation

Constraint

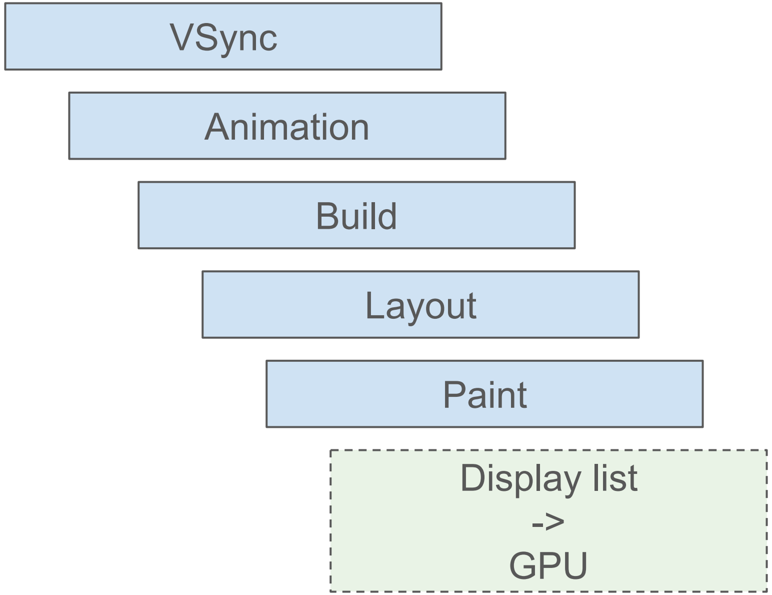
Layout

Display

Display list
->
GPU

Flutter的渲染阶段

Flutter’s Rendering Pipeline



VSync

Animation

Build

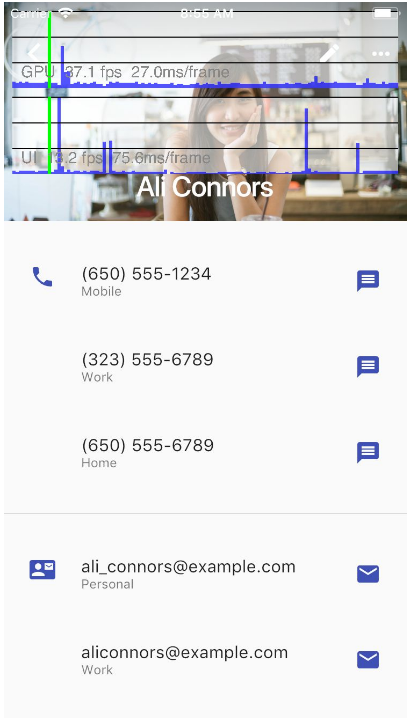
Layout

Paint

Display list
->
GPU

检测Flutter的渲染

Instrumenting Flutter’s Rendering



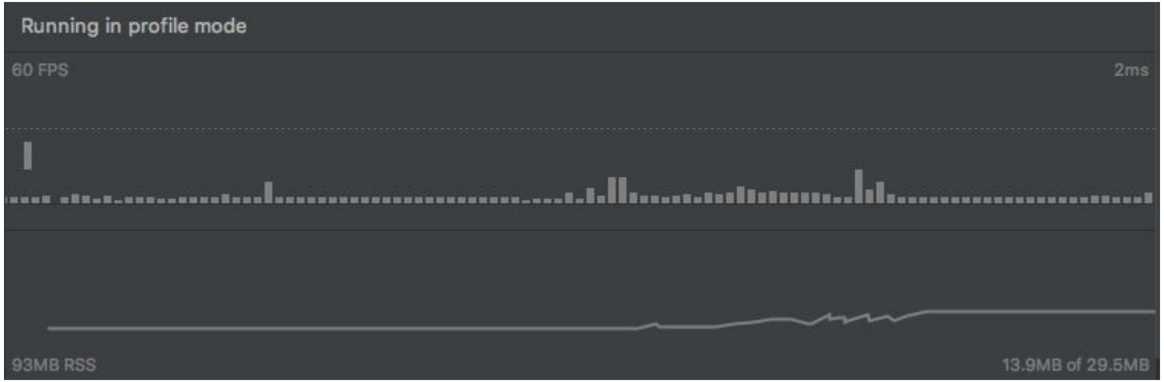
(650) 555-1234
Mobile

(323) 555-6789
Work

(650) 555-6789
Home

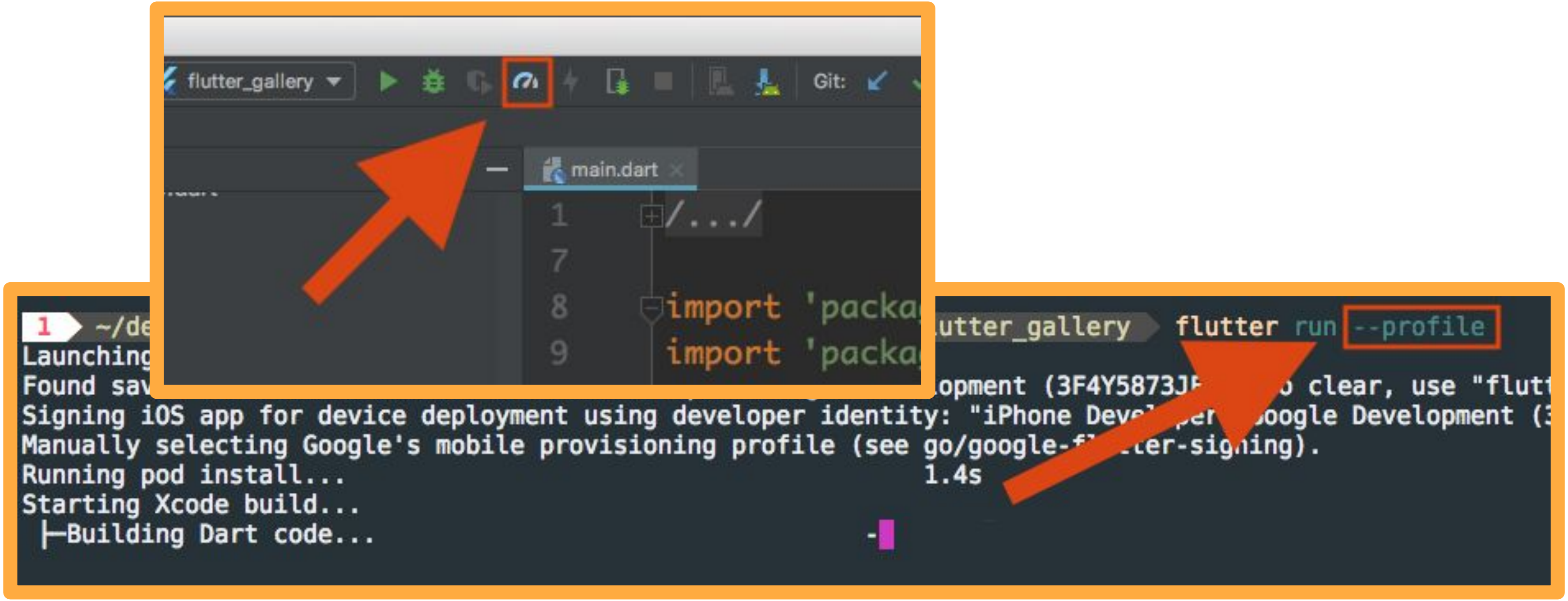
ali_connors@example.com
Personal

aliconnors@example.com
Work



Profile模式

Profile Mode



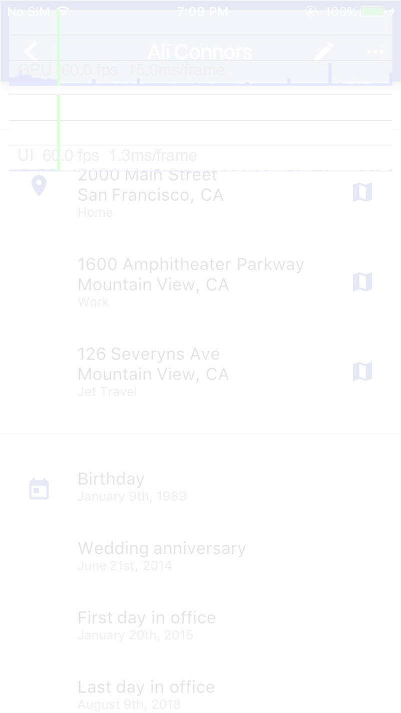
```
/.../

import 'packa
import 'packa
```

```
1  ~/de                                          utter_gallery  flutter run --profile
Launching
Found sav                                        opment (3F4Y5873J         o clear, use "flutt
Signing iOS app for device deployment using developer identity: "iPhone Deve  per  oogle Development (3
Manually selecting Google's mobile provisioning profile (see go/google-fl  ter-signing).
Running pod install...                                     1.4s
Starting Xcode build...
 ├─Building Dart code...                                 -
```

检测Flutter的渲染

Instrumenting Flutter’s Rendering



2000 Main Street
San Francisco, CA
Home

1600 Amphitheater Parkway
Mountain View, CA
Work

126 Severyns Ave
Mountain View, CA
Jet Travel

Birthday
January 9th, 1989

Wedding anniversary
June 21st, 2014

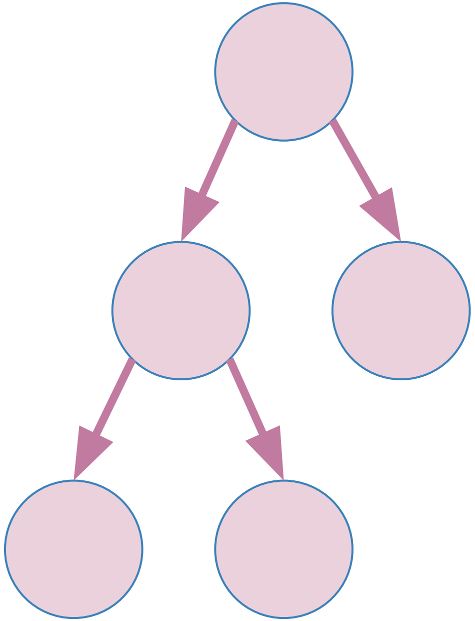
First day in office
January 20th, 2015

Last day in office
August 9th, 2018

DEMO

渲染责任部件

Rendering Actors

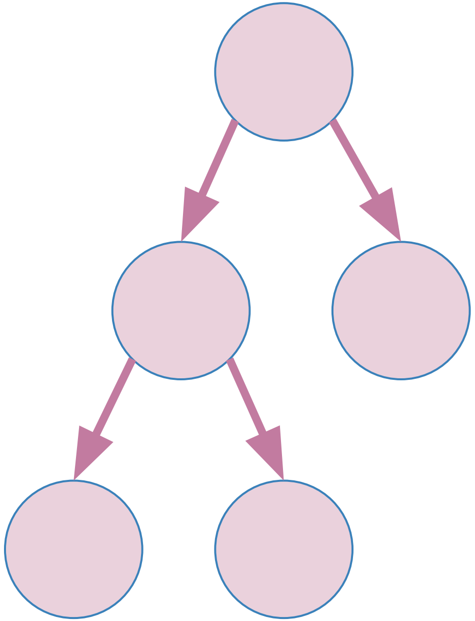


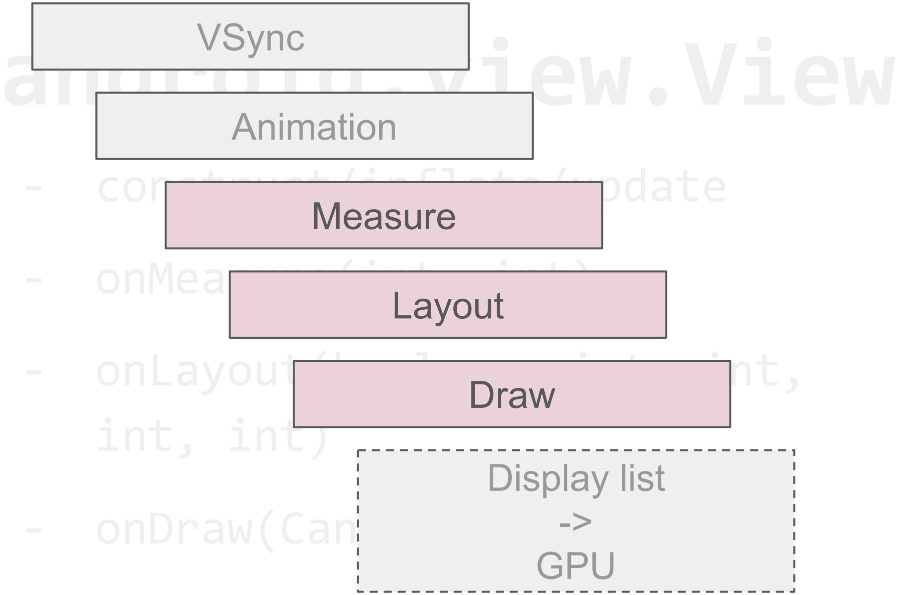
android.view.View

- construct/inflate/update
- onMeasure(int, int)
- onLayout(boolean, int, int, int, int)
- onDraw(Canvas)

渲染责任部件

Rendering Actors



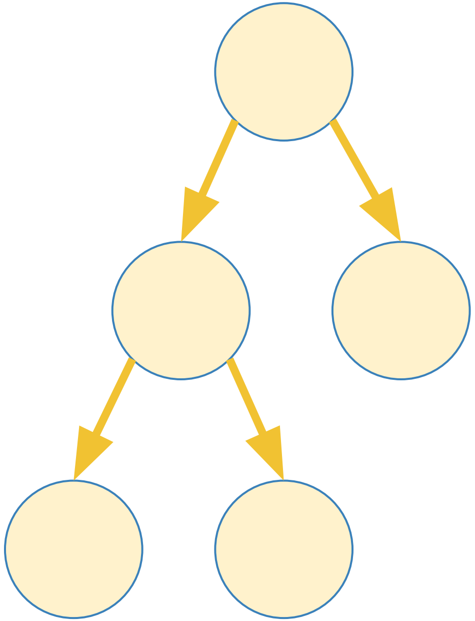


android.view.View

- construct/inflate/update
- onMeasure(int, int)
- onLayout(boolean, int, int, int, int)
- onDraw(Canvas)

渲染责任部件

Rendering Actors

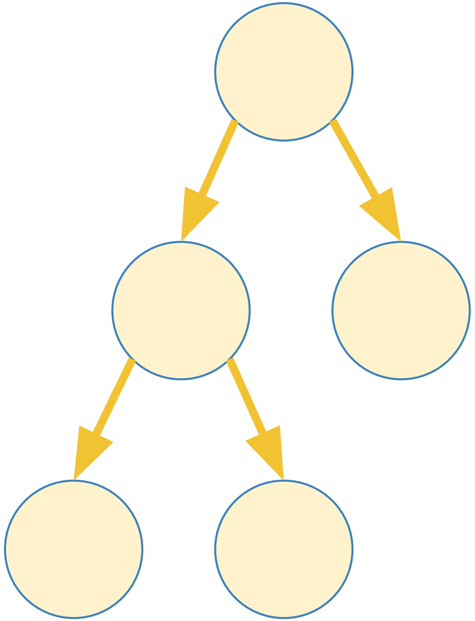


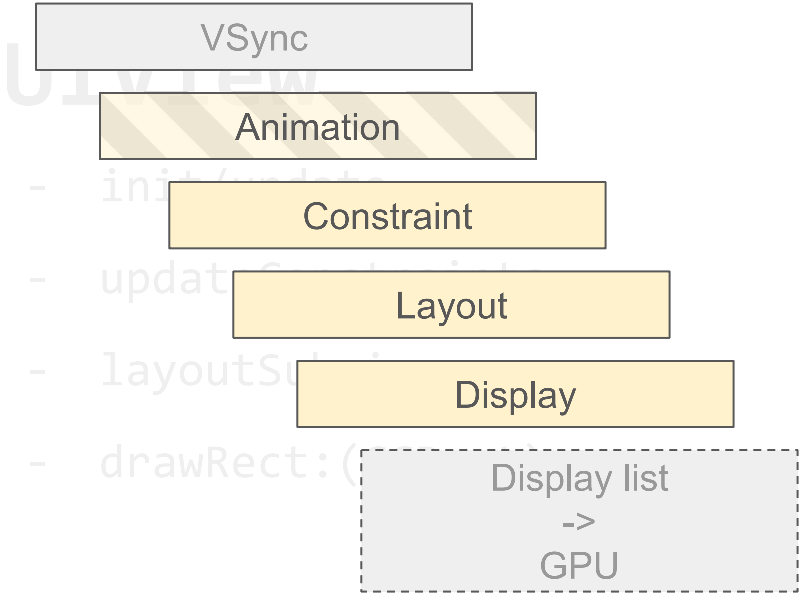
UIView

- init/update
- updateConstraints
- layoutSubviews
- drawRect:(CGRect)

渲染责任部件

Rendering Actors





VSync

Animation

Constraint

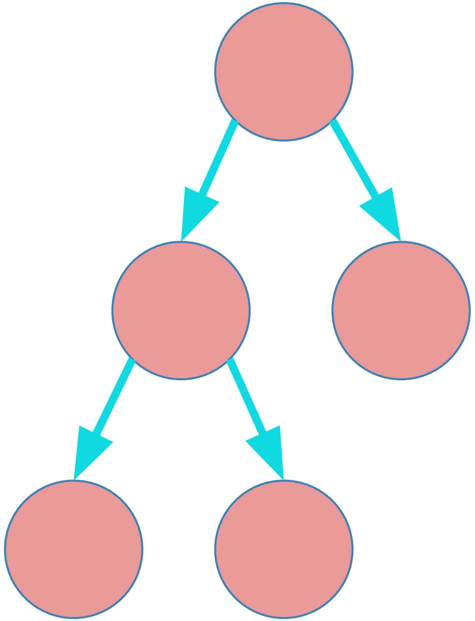
Layout

Display

Display list
->
GPU

Flutter的渲染责任部件

Flutter’s Rendering Actors

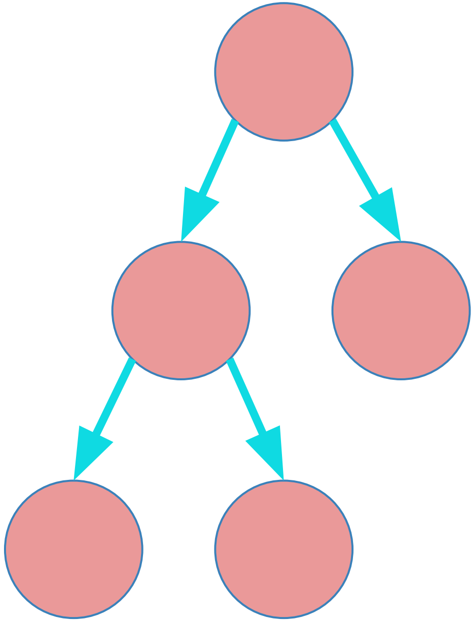


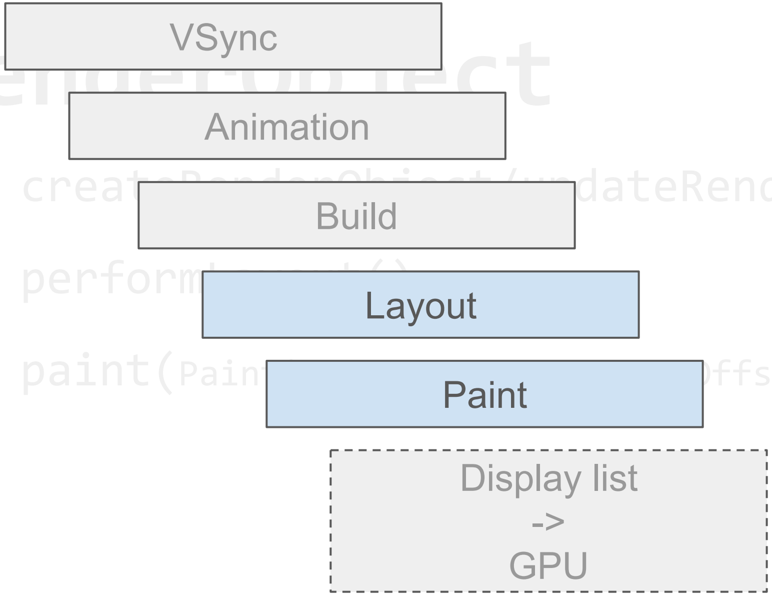
RenderObject

- createRenderObject/updateRenderObject
- performLayout()
- paint(PaintingContext context, Offset offset)

Flutter的渲染责任部件

Flutter’s Rendering Actors





VSync

Animation

Build

Layout

Paint

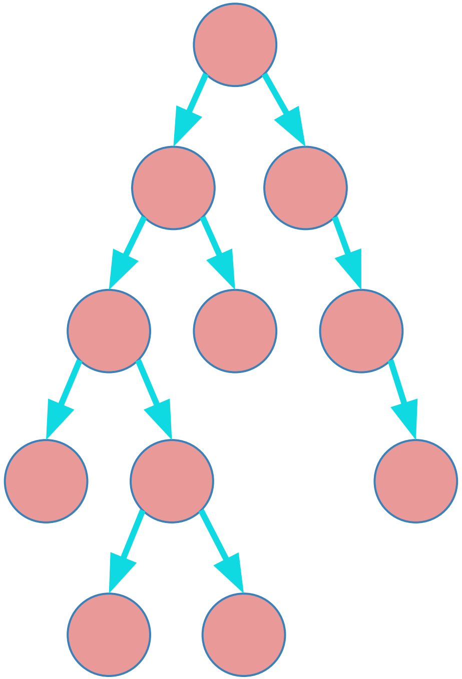
Display list
->
GPU

RenderObject

- createRenderObject/updateRenderObject
- performLayout()
- paint(PaintingContext context, Offset offset)

Flutter的渲染责任部件

Flutter’s Rendering Actors

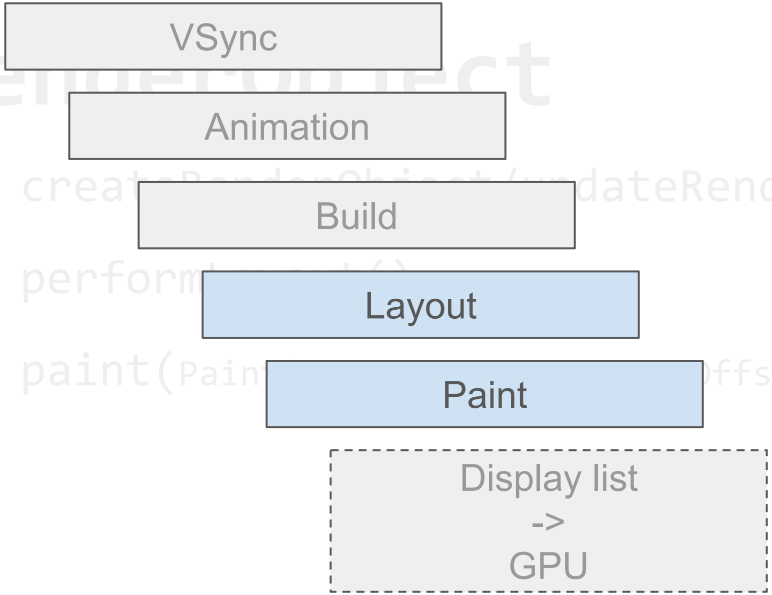


Re ect

- create dateRenderObject

- perform

- paint(Pain ffset offset)



VSync

Animation

Build

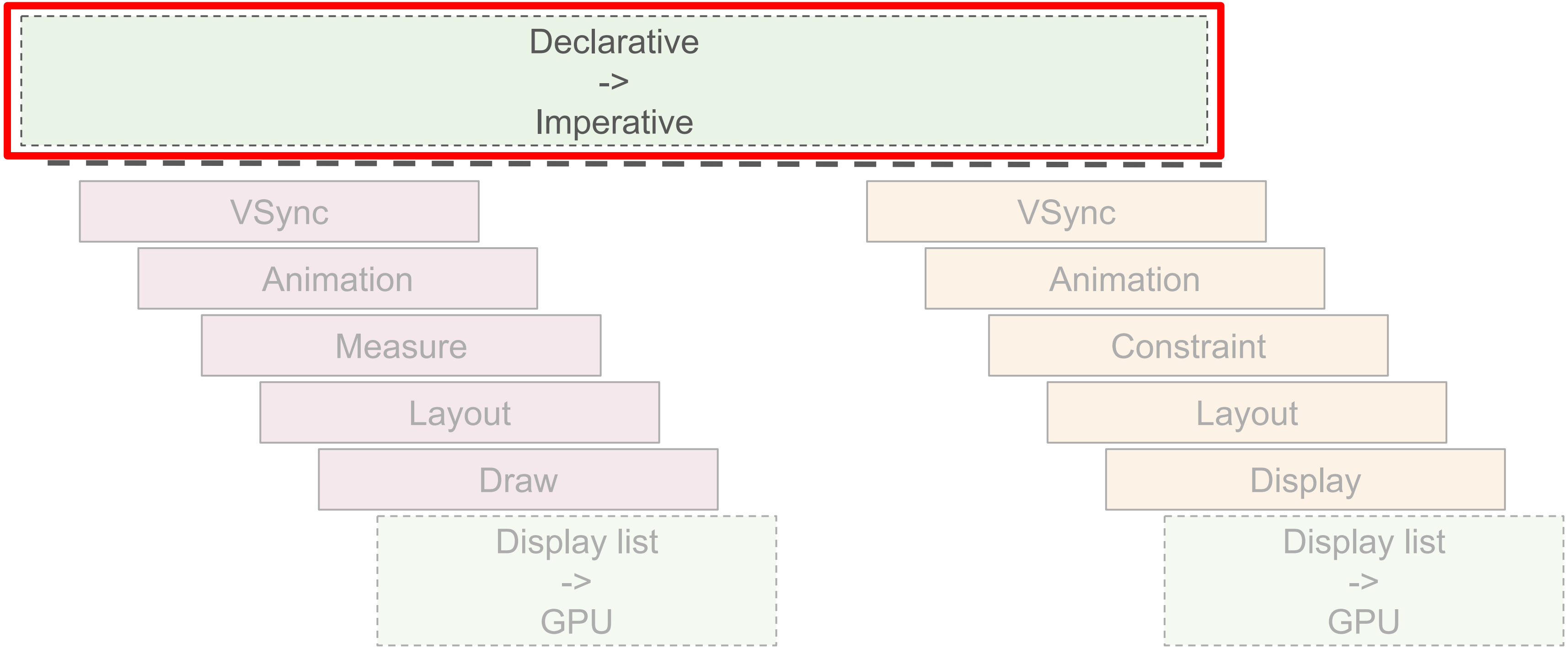
Layout

Paint

Display list
->
GPU

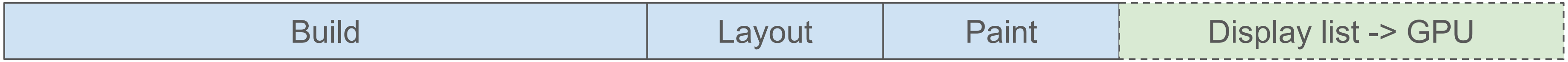
Flutter的声明式UI

Flutter’s Declarative UI



Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases



Build

Layout

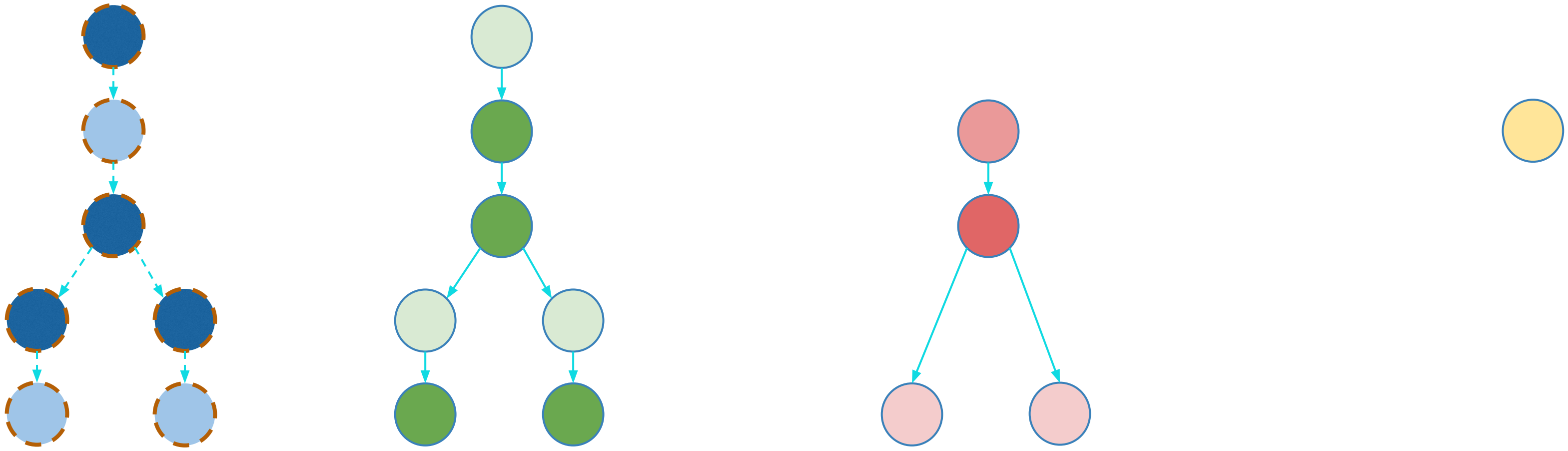
Paint

Display list -> GPU

Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

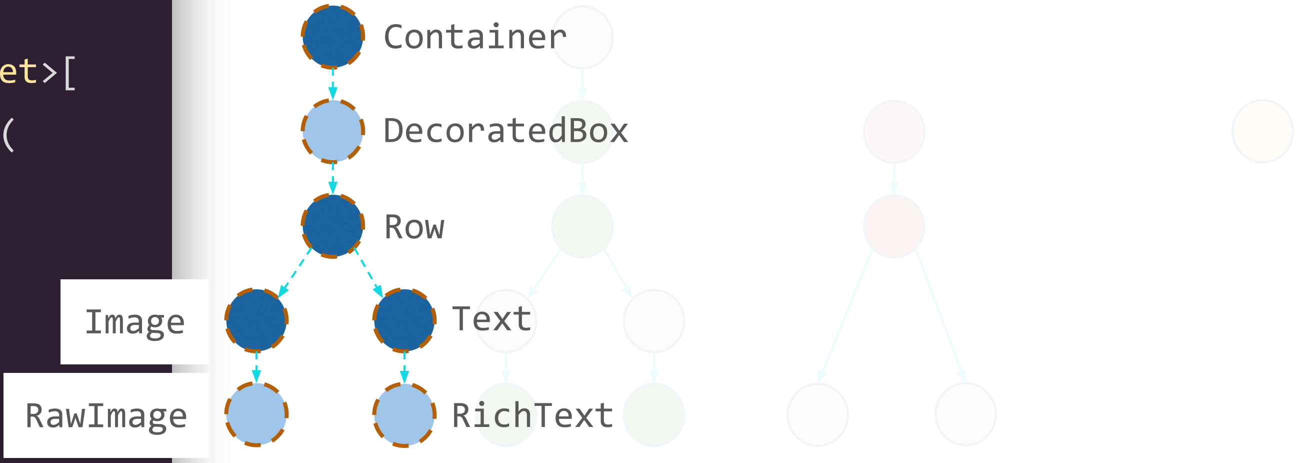


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

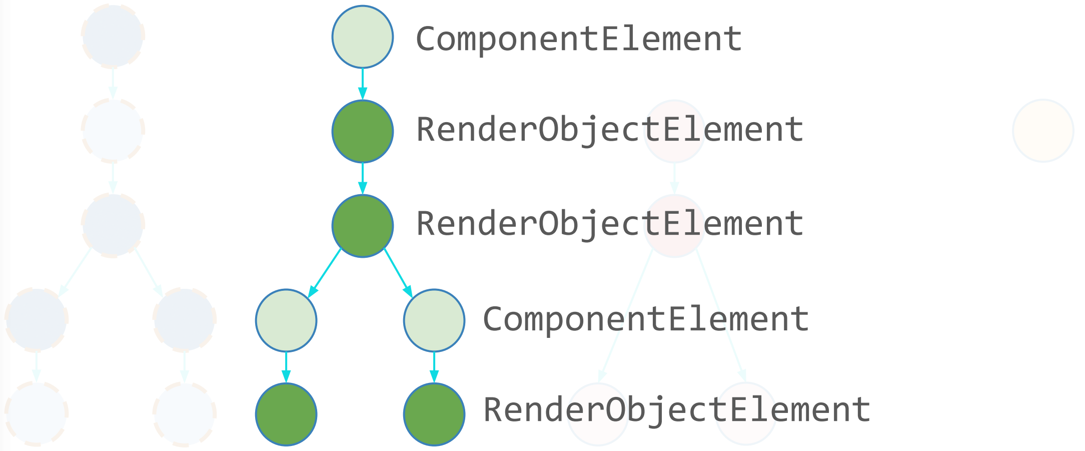


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

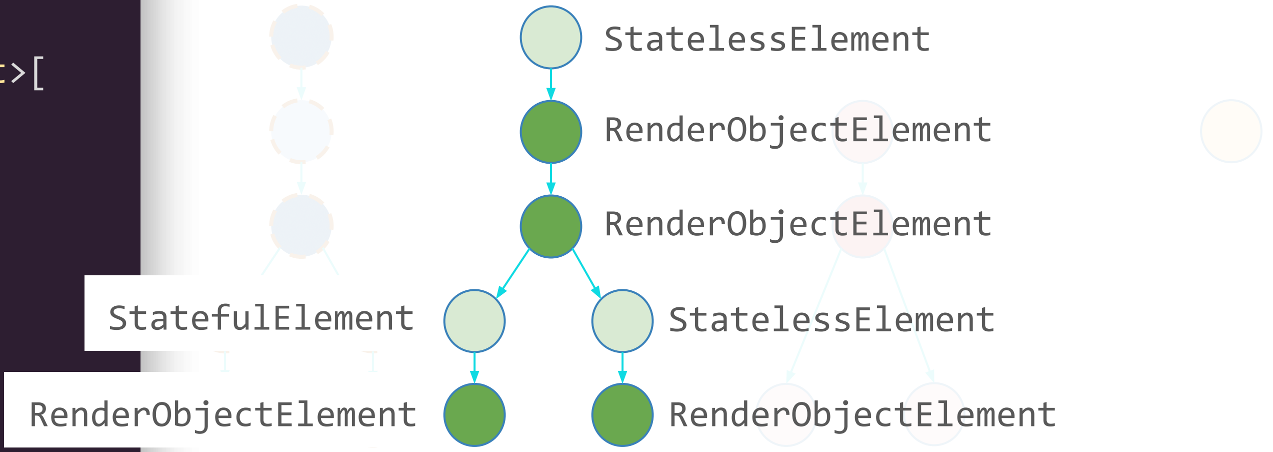


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

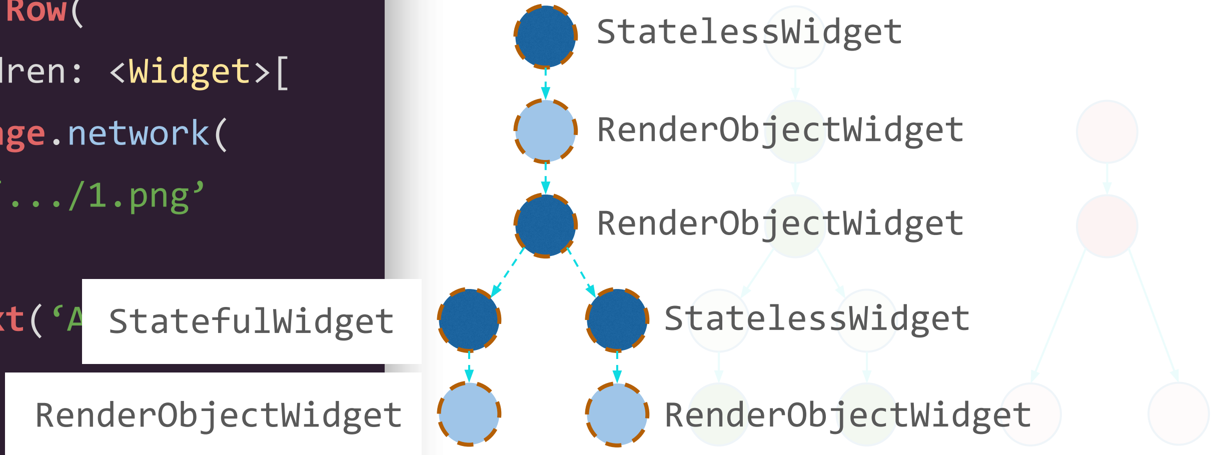


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A
    ],
  ),
)
```

<table>
<tr><td colspan="2">Build</td><td>Layout</td><td>Paint</td><td>Display list -> GPU</td></tr>
<tr><td>Widget</td><td>Element</td><td colspan="2">RenderObject</td><td>Layer</td></tr>
</table>

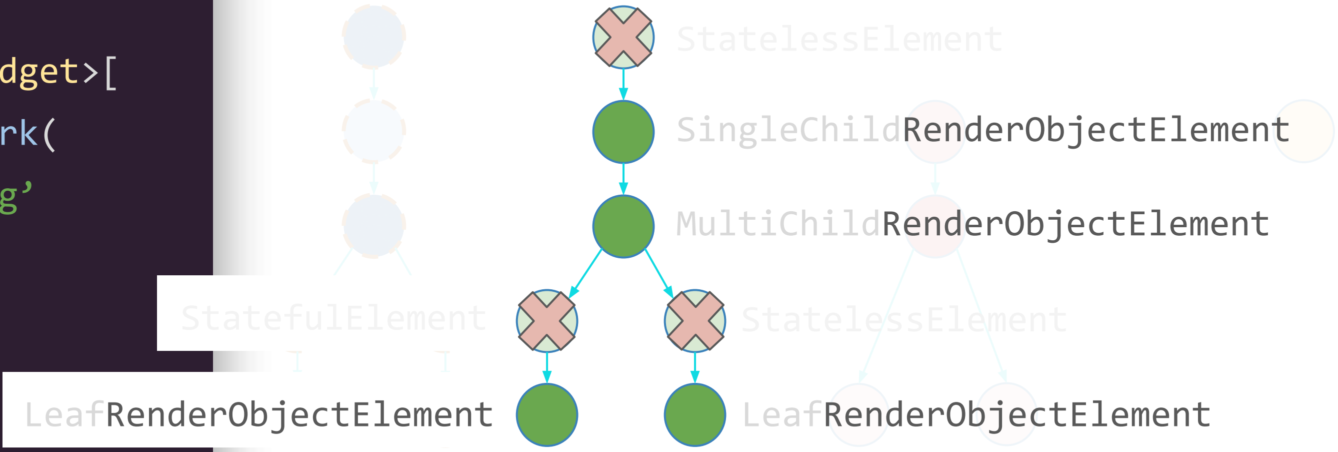


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

<table>
<tr><td colspan="2">Build</td><td>Layout</td><td>Paint</td><td>Display list -> GPU</td></tr>
<tr><td>Widget</td><td>Element</td><td colspan="2">RenderObject</td><td>Layer</td></tr>
</table>

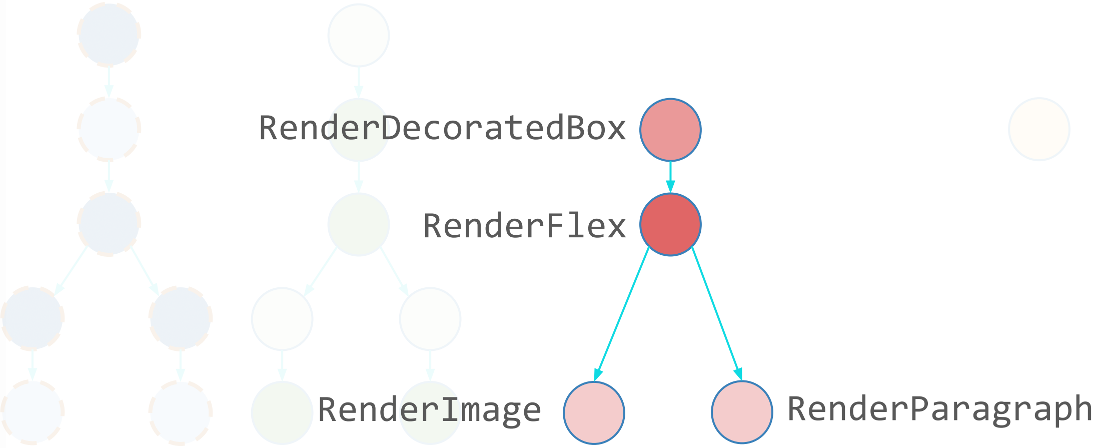


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

<table>
<tr><td colspan="2">Build</td><td>Layout</td><td>Paint</td><td>Display list -> GPU</td></tr>
<tr><td>Widget</td><td>Element</td><td colspan="2">RenderObject</td><td>Layer</td></tr>
</table>

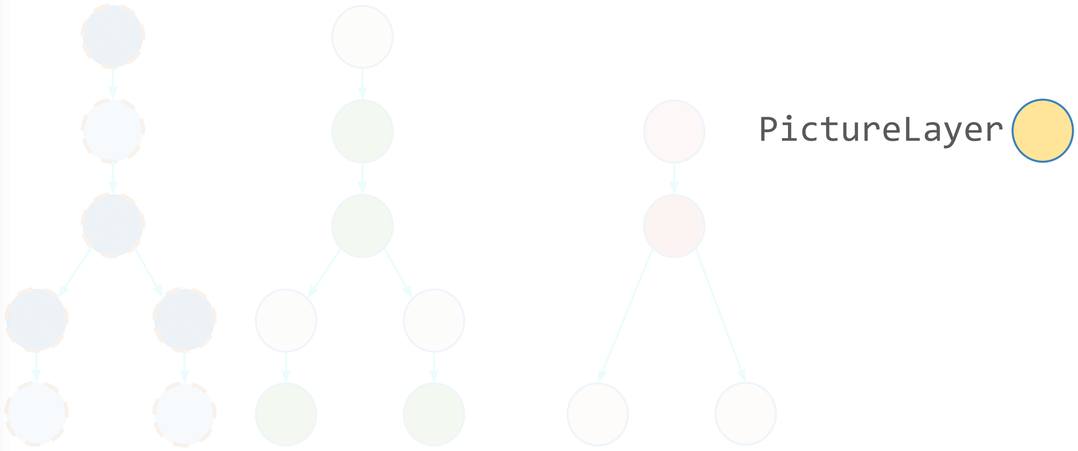


Flutter的树与渲染阶段

Flutter’s Trees and Rendering Phases

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

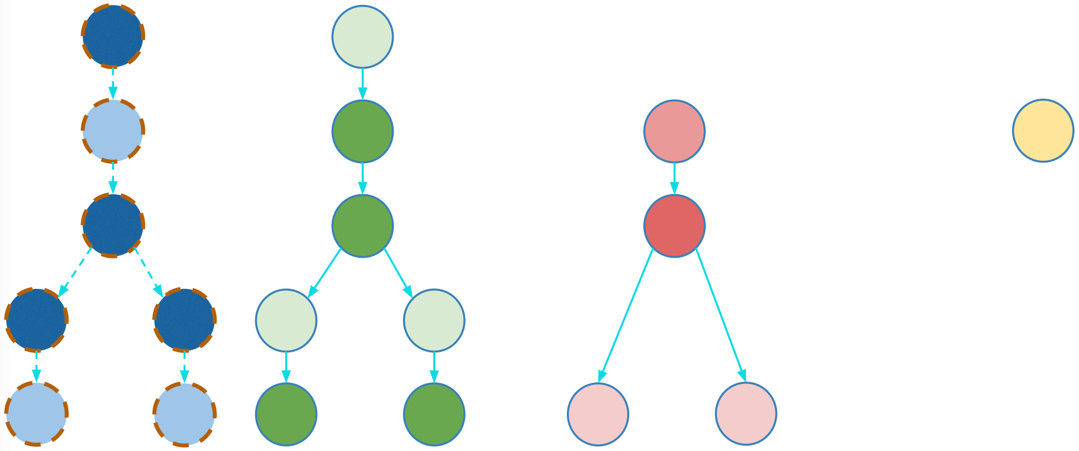


树工程

Tree Operations

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘A’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer



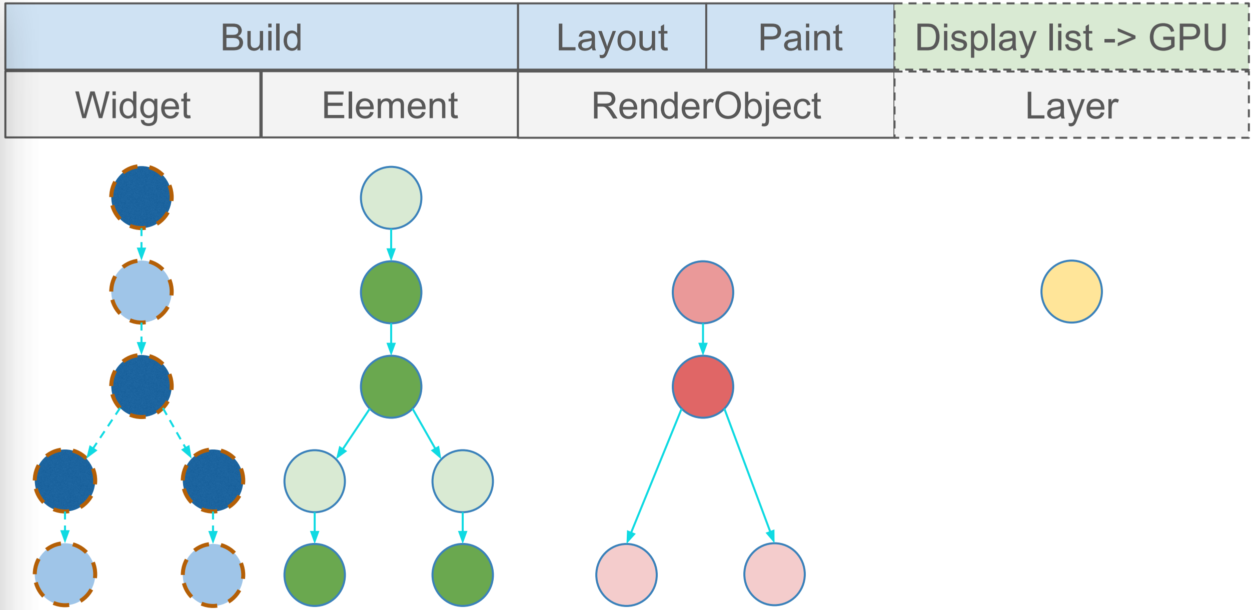
了解build阶段

Understanding the Build Phase

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```

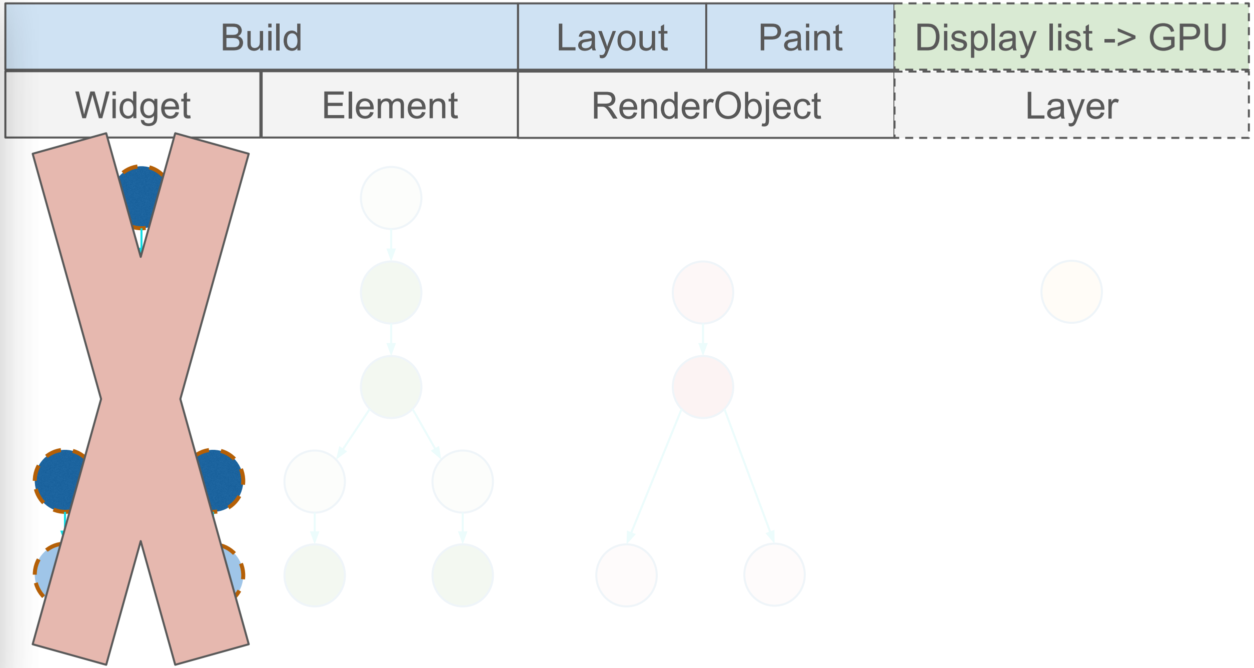


Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        '.../1.png'
      ),
      Text('B'),
    ],
  ),
)
```



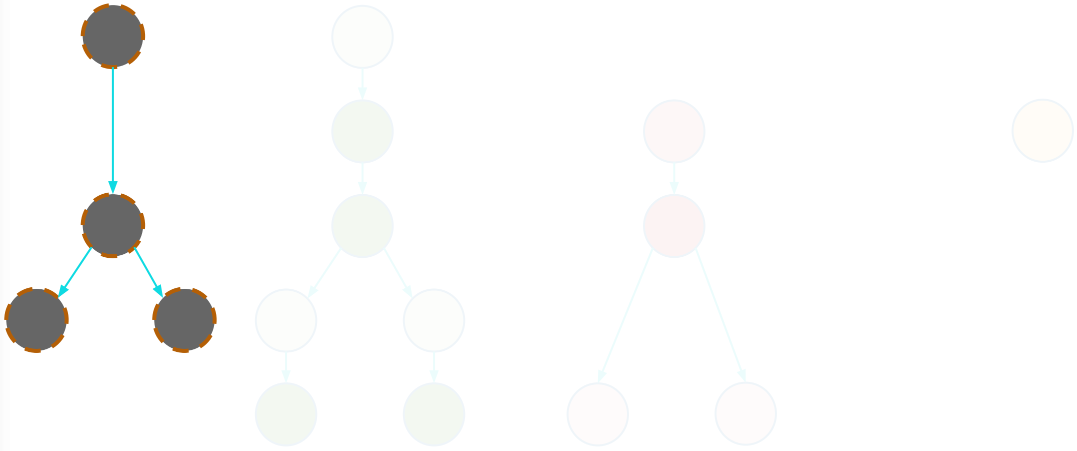
Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

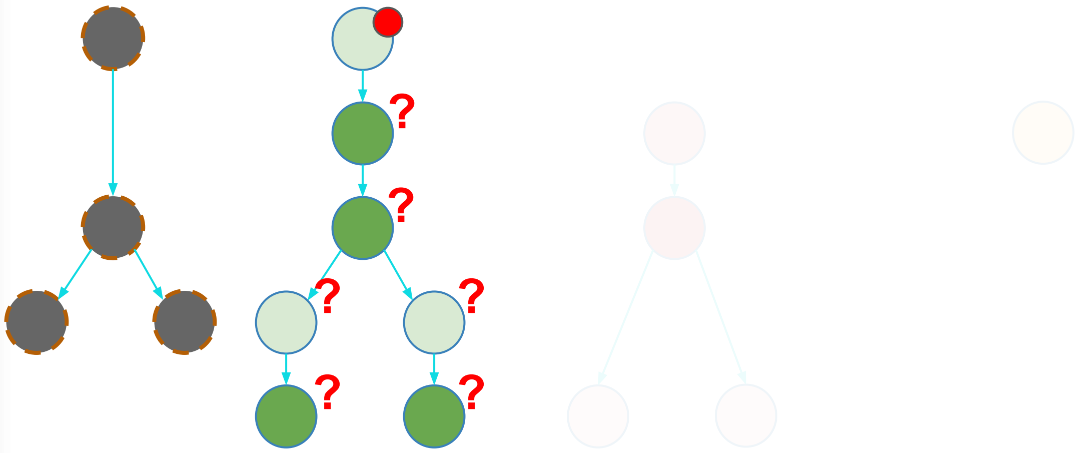


同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```

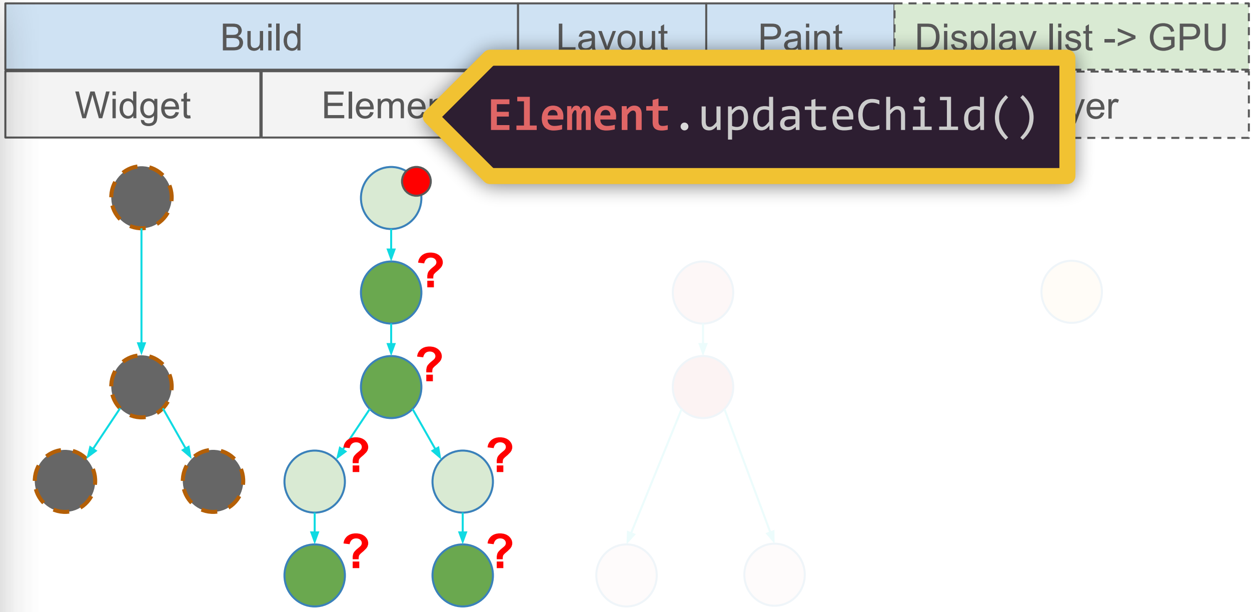
Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer



同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```

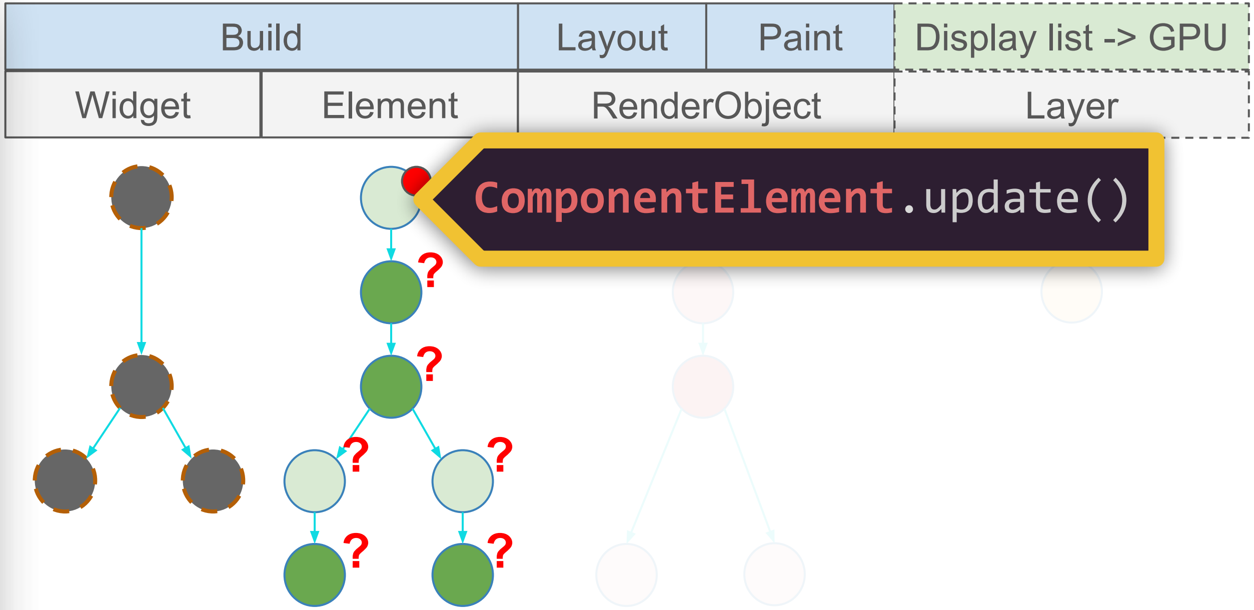


Element.updateChild()

同类型更新

Same Type Update

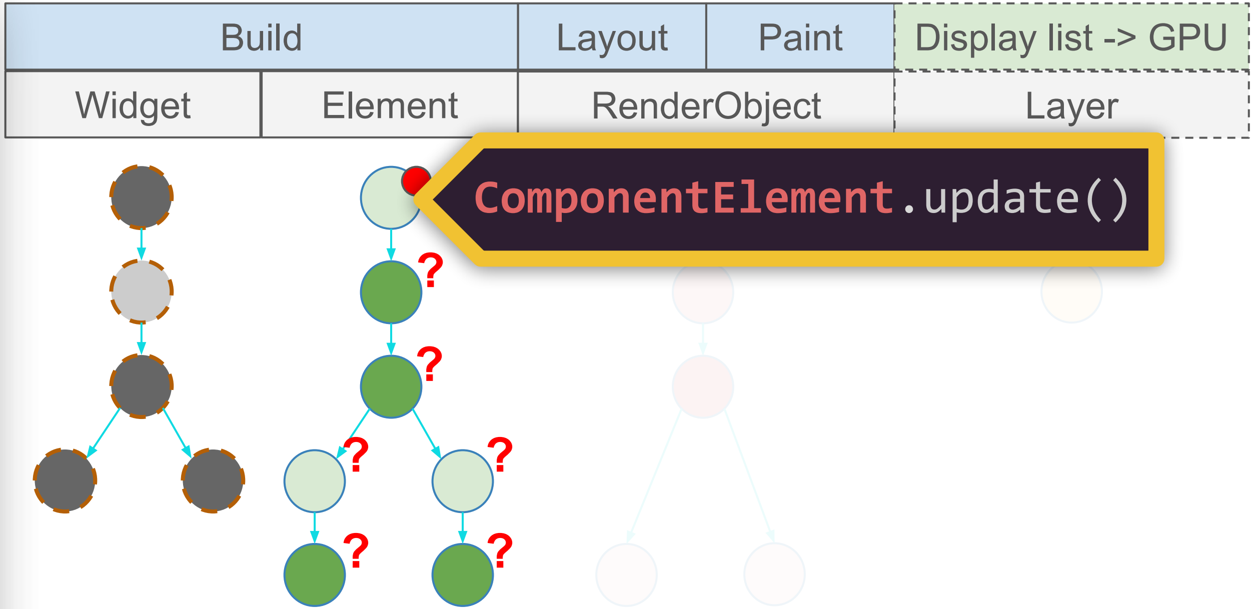
```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



同类型更新

Same Type Update

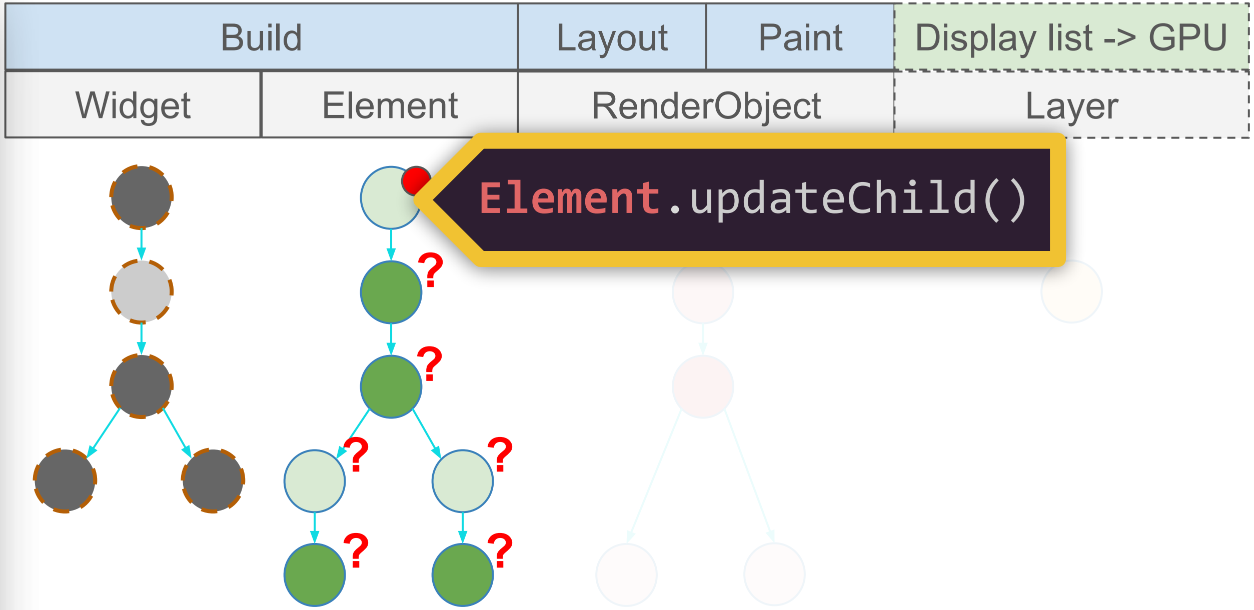
```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        '.../1.png'
      ),
      Text('B'),
    ],
  ),
)
```



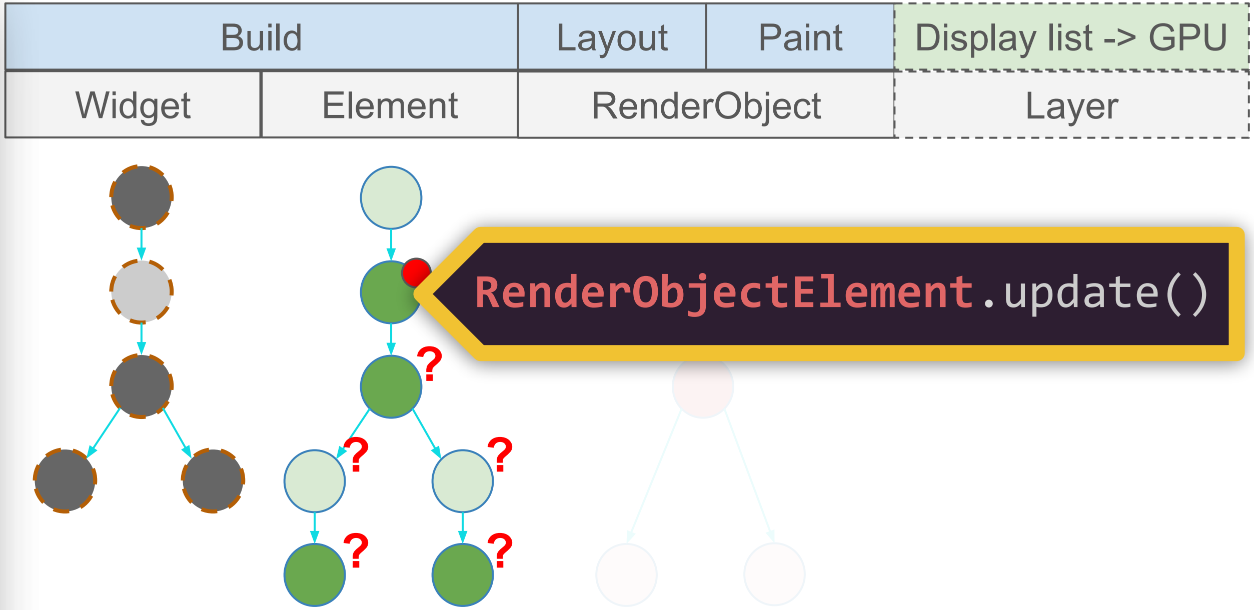
Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

Element.updateChild()

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



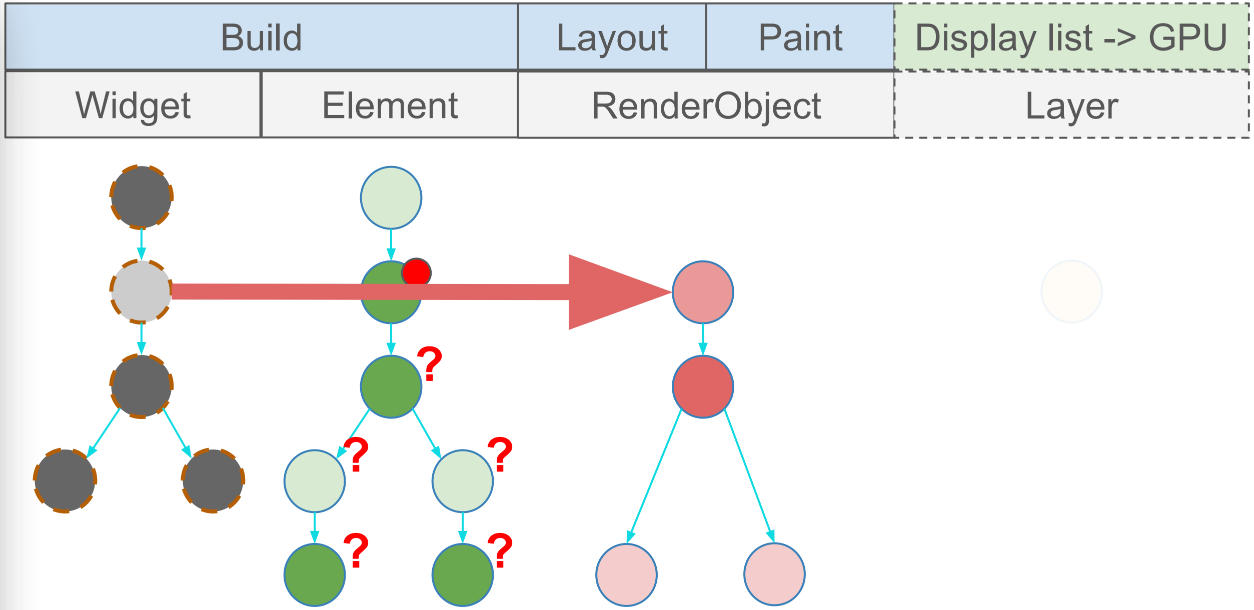
Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

RenderObjectElement.update()

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

同类型更新

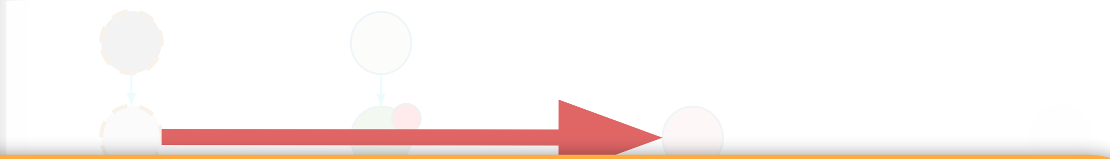
Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(



    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

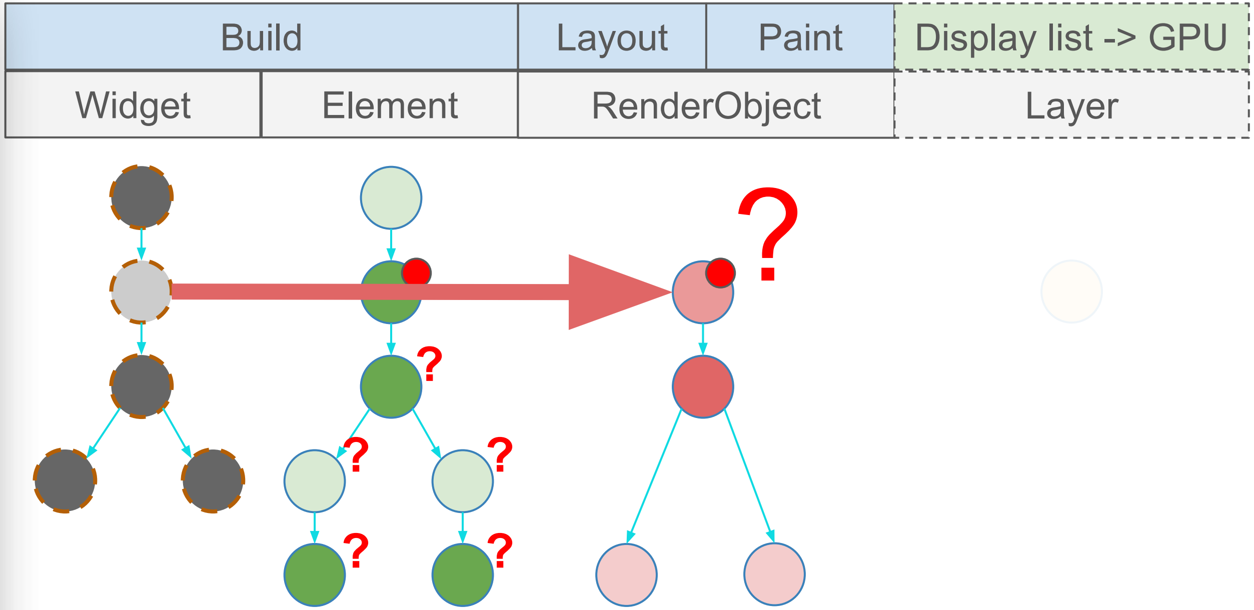


```
@override
void updateRenderObject(BuildContext context, RenderDecoratedBox renderObject) {
  renderObject
    ..decoration = decoration
    ..configuration = createLocalImageConfiguration(context)
    ..position = position;
}
```

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

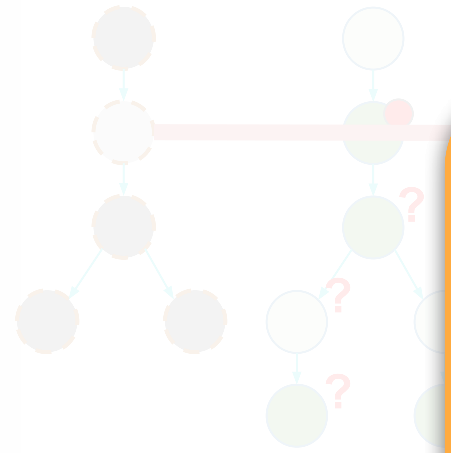
```
set decoration(Decoration value) {
  assert(value != null);
  if (value == _decoration)
    return;
  _painter?.dispose();
  _painter = null;
  _decoration = value;
  markNeedsPaint();
}
```

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

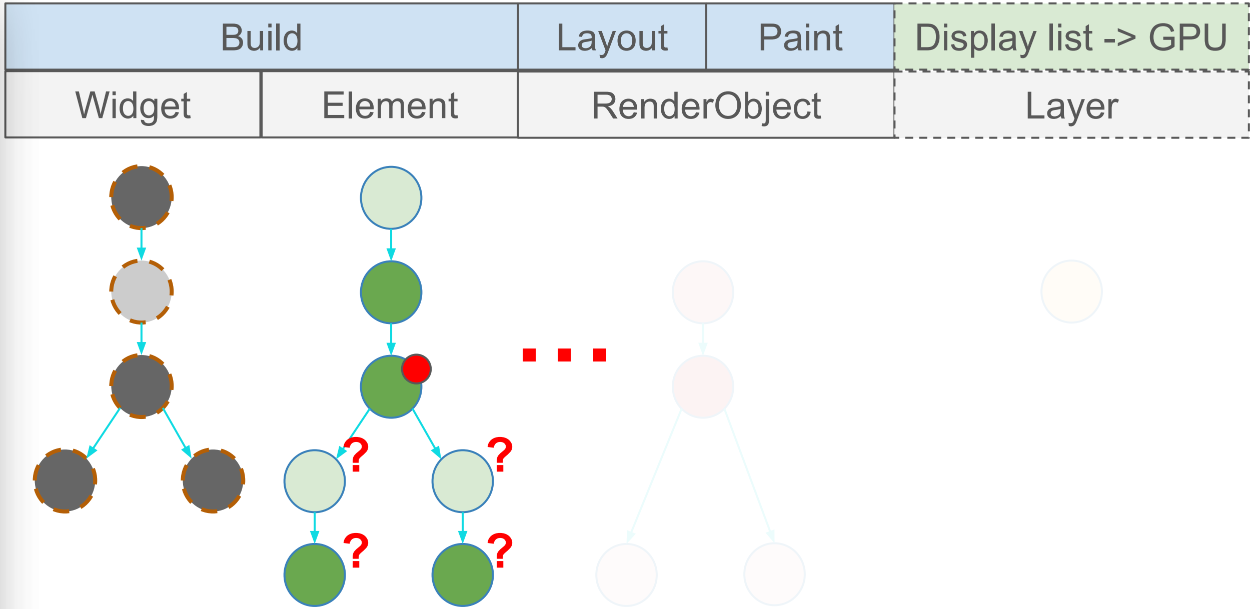


```
set decoration(Decoration value) {
  assert(value != null);
  if (value == _decoration)
    return;
  _painter?.dispose();
  _painter = null;
  _decoration = value;
  markNeedsPaint();
}
```

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



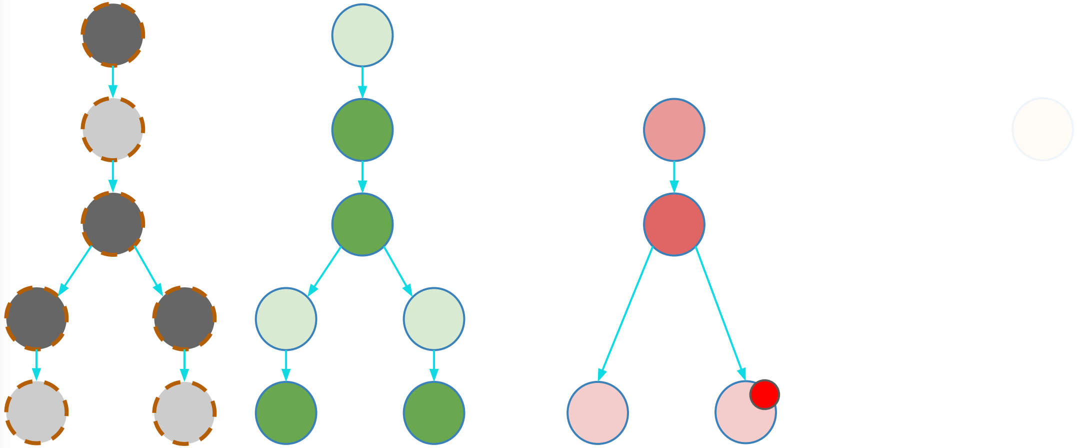
Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

同类型更新

Same Type Update

```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        '.../1.png'
      ),
      Text('B'),
    ],
  ),
)
```

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer



如何提高build效率

How to Improve Build Efficiency

调试工具

Performance Debugging Tools

- debugPrintBeginFrameBanner/debugPrintEndFrameBanner
  - 每帧开始/结束
- debugPrintScheduleBuildForStacks
  - 为什么被构建
- debugPrintRebuildDirtyWidgets
  - 什么组件被重新构建了
- debugProfileBuildsEnabled
  - 在观测台里显示构建树

调试工具

Performance Debugging Tools

- debugPrintBeginFrameBanner/debugPrintEndFrameBanner
  - 每帧开始/结束
- debugPrintScheduleBuildForStacks
  - 为什么被构建
- debugPrintRebuildDirtyWidgets
  - 什么组件被重构建了
- debugProfileBuildsEnabled
  - 在观测台里显示构建树

调试工具

Performance Debugging Tools

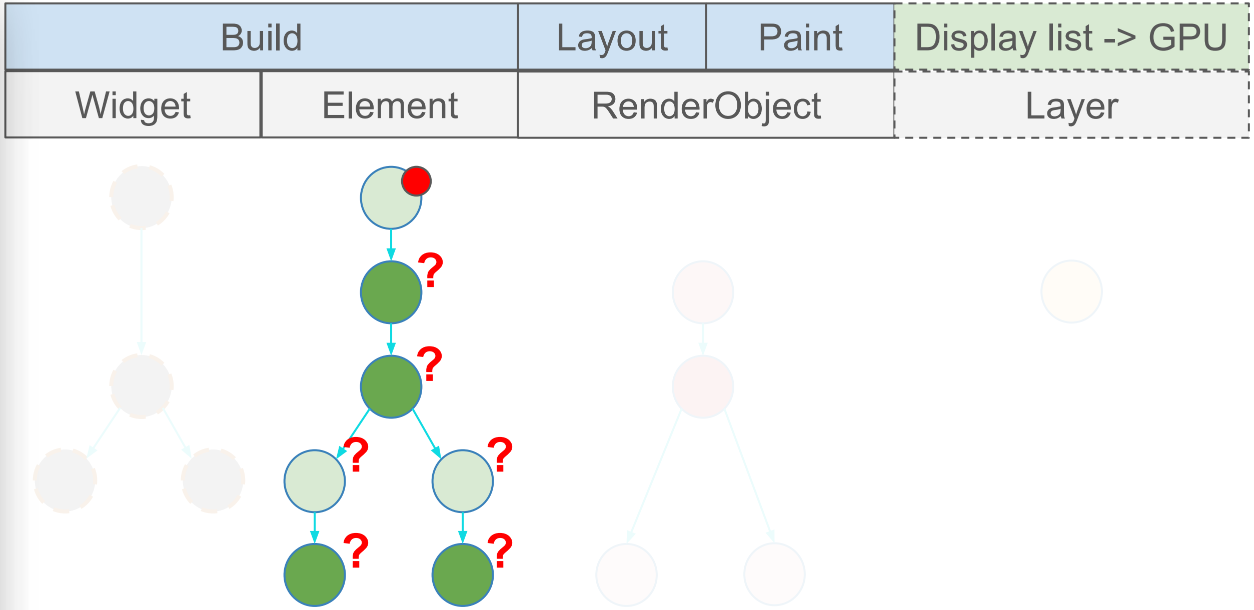
- debugPrintBeginFrameBanner/debugPrintEndFrameBanner
  - 每帧开始/结束
- debugPrintScheduleBuildForStacks
  - 为什么被构建
- debugPrintRebuildDirtyWidgets
  - 什么组件被重构建了
- debugProfileBuildsEnabled
  - 在观测台里显示构建树

DEMO

如何提高build效率

How to Improve Build Efficiency

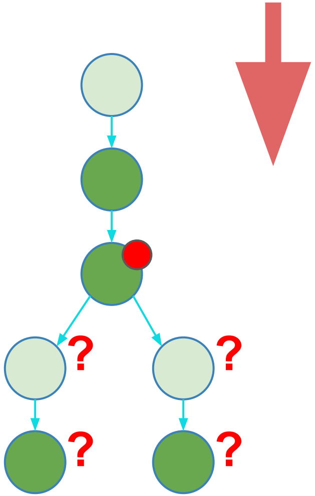
```
Container(
  color: Colors.blue,
  child: Row(
    children: <Widget>[
      Image.network(
        ‘.../1.png’
      ),
      Text(‘B’),
    ],
  ),
)
```



Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer

如何提高build效率

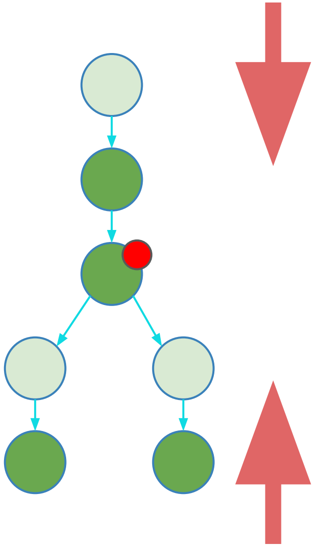
How to Improve Build Efficiency



降低遍历的出发点

如何提高build效率

How to Improve Build Efficiency

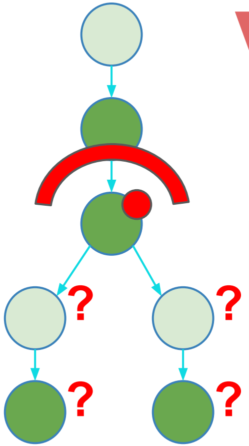


降低遍历的出发点

停止树的遍历

如何提高build效率

How to Improve Build Efficiency



↓降低遍历的出发点

1. setState
2. InheritedWidget
3. 热重载

如何提高build效率

How to Improve Build Efficiency

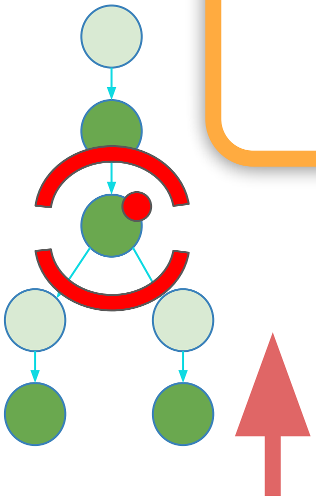
降低遍历的出发点

DEMO

1. setState
2. InheritedWidget
3. 热重载

如何提高build效率

How to Improve Build Efficiency

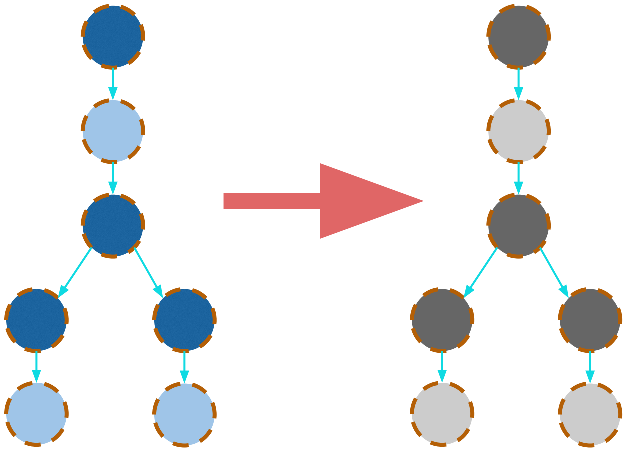


1. 重用同一子组件实例

停止树的遍历

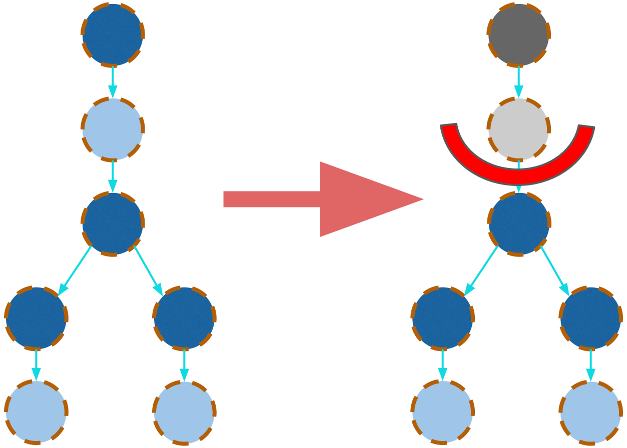
如何提高build效率

How to Improve Build Efficiency



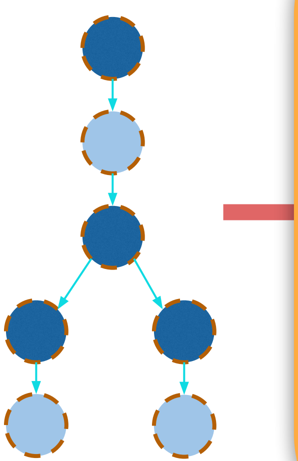
如何提高build效率

How to Improve Build Efficiency



如何提高build效率

How to Improve Build Efficiency



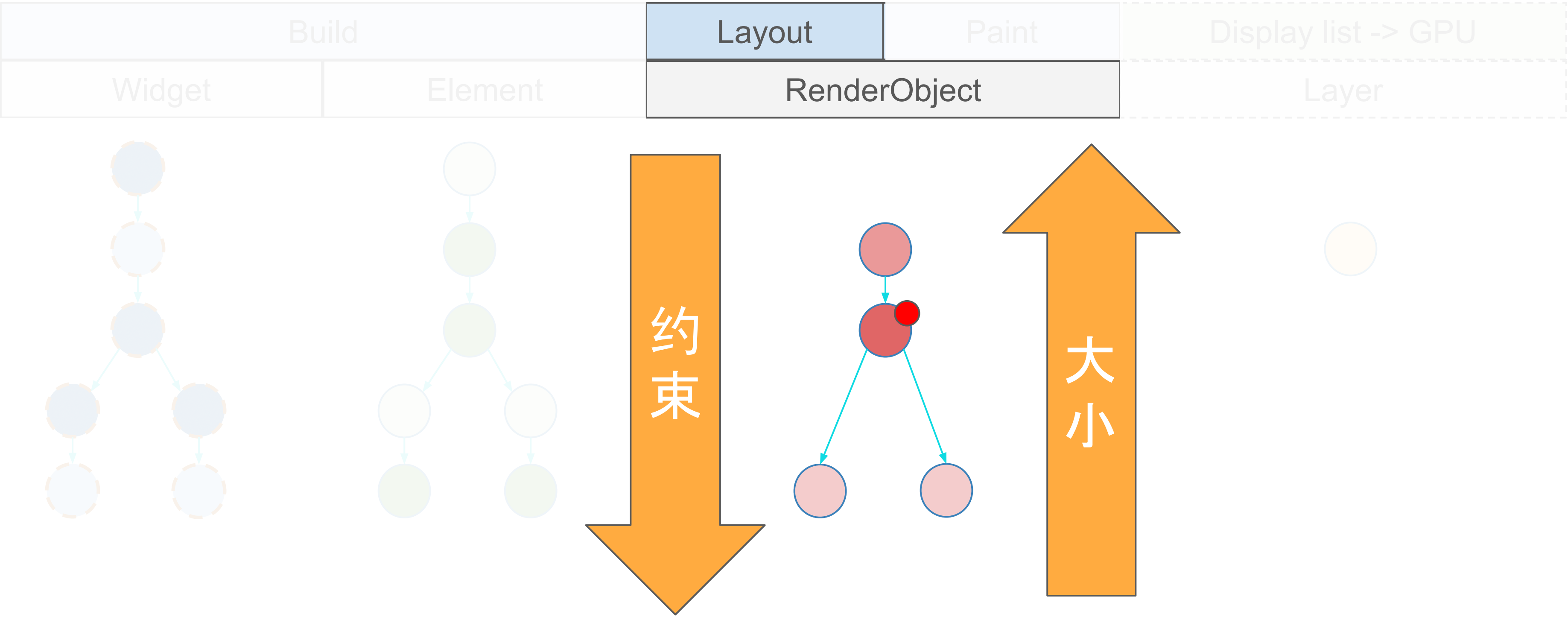
```
SlideTransition(
   ...
   child: Row(...)
)
```

了解layout阶段

Understanding the Layout Phase

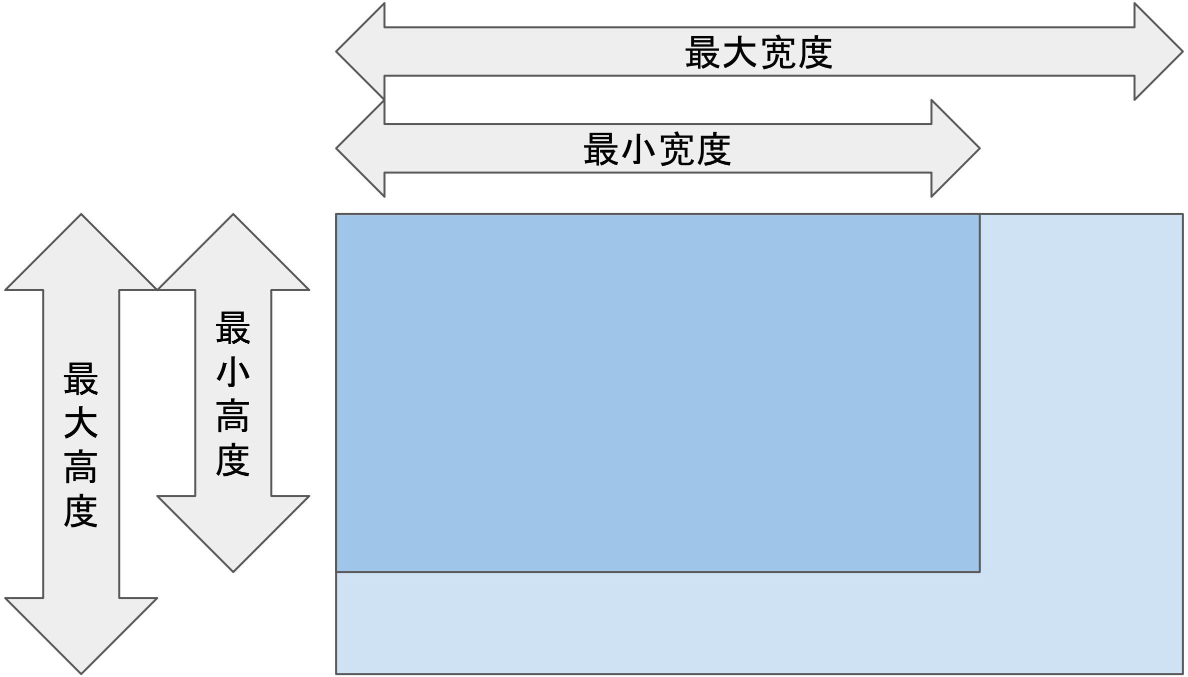
布局阶段

Layout Phase



布局阶段

Layout Phase

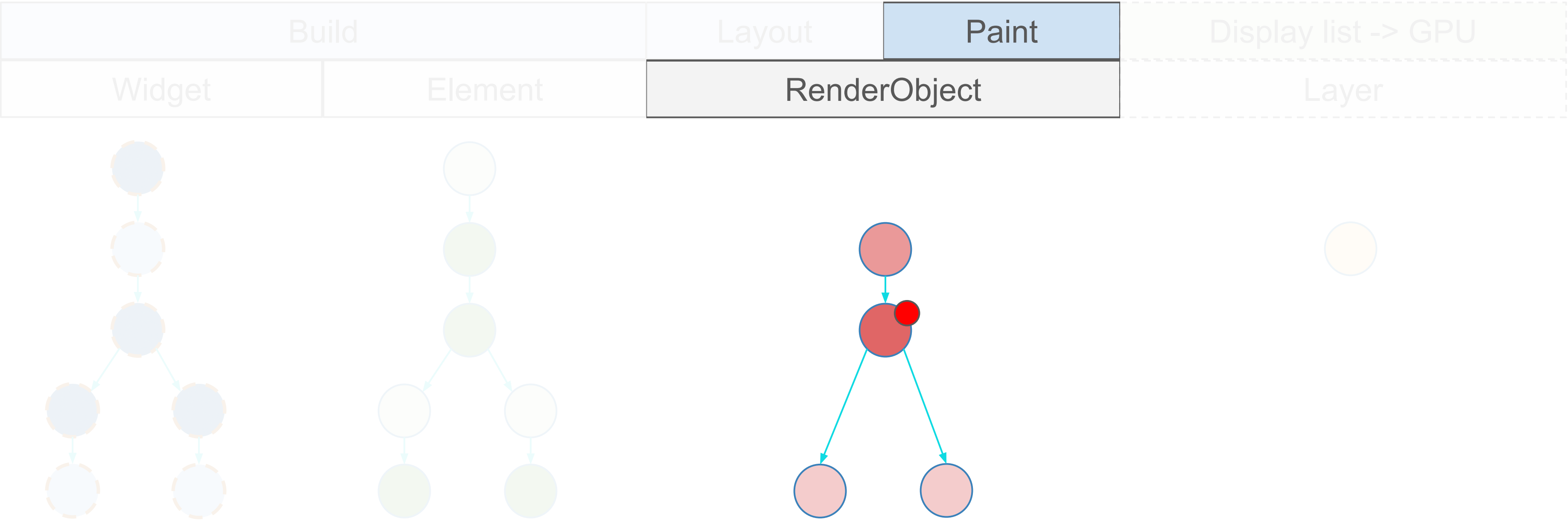


了解paint阶段

Understanding the Paint Phase

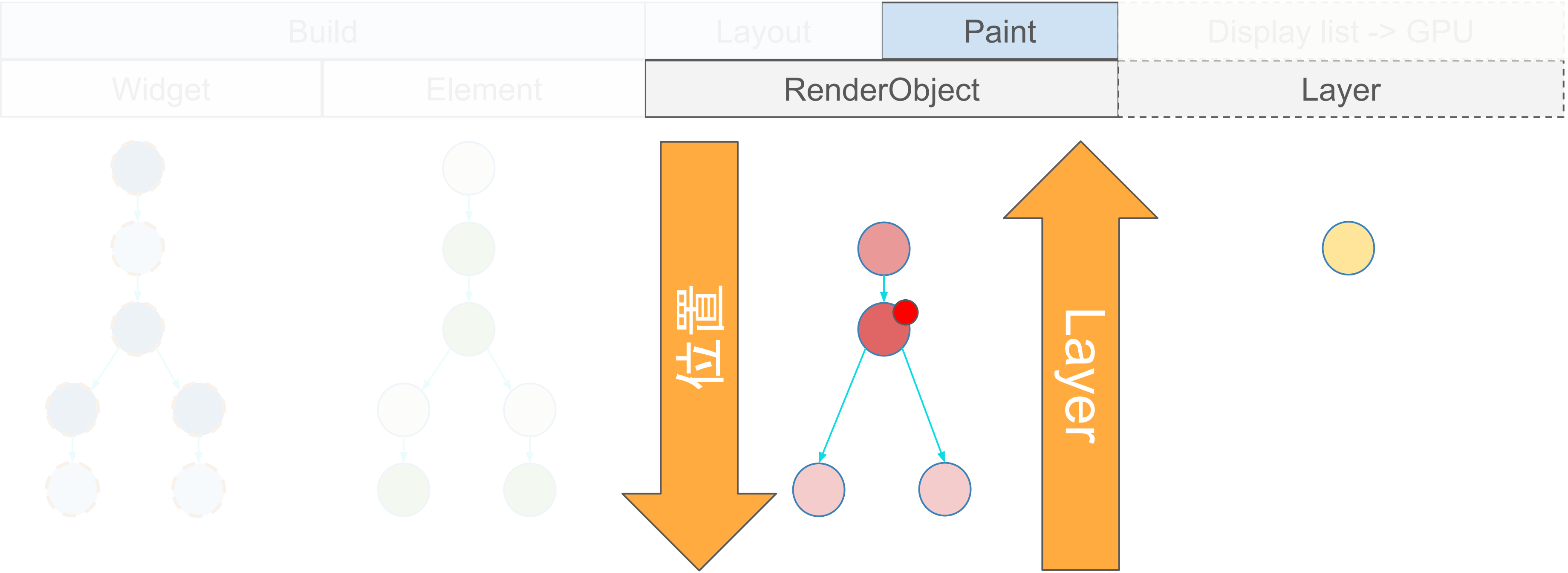
绘制阶段

Paint Phase



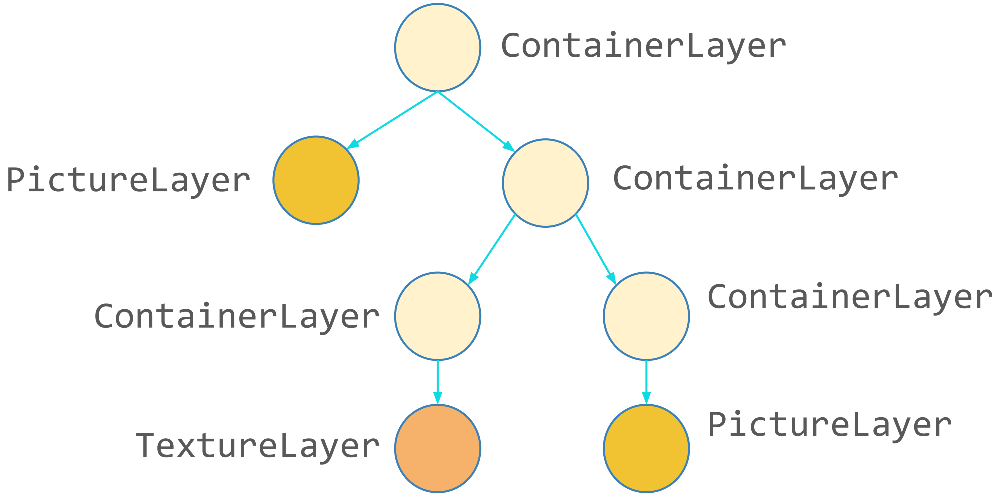
绘制阶段

Paint Phase



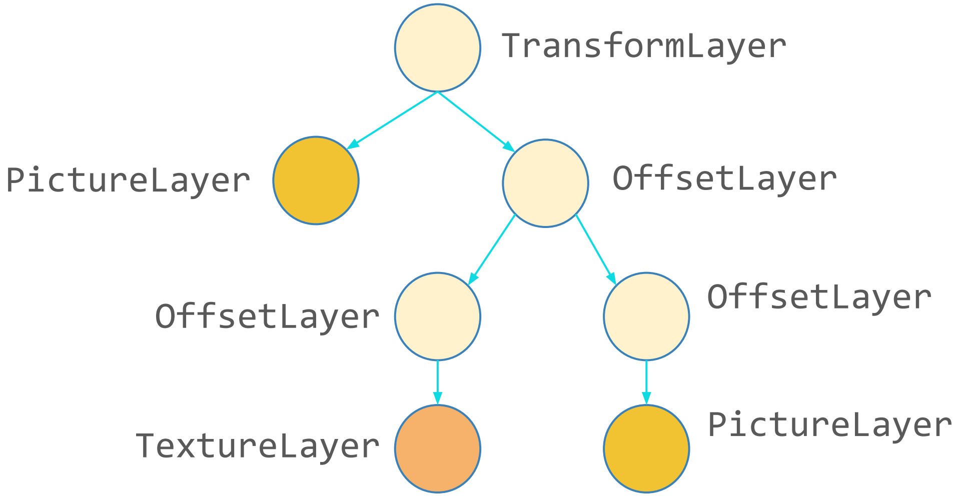
Layer种类

Layer Types



Layer种类

Layer Types

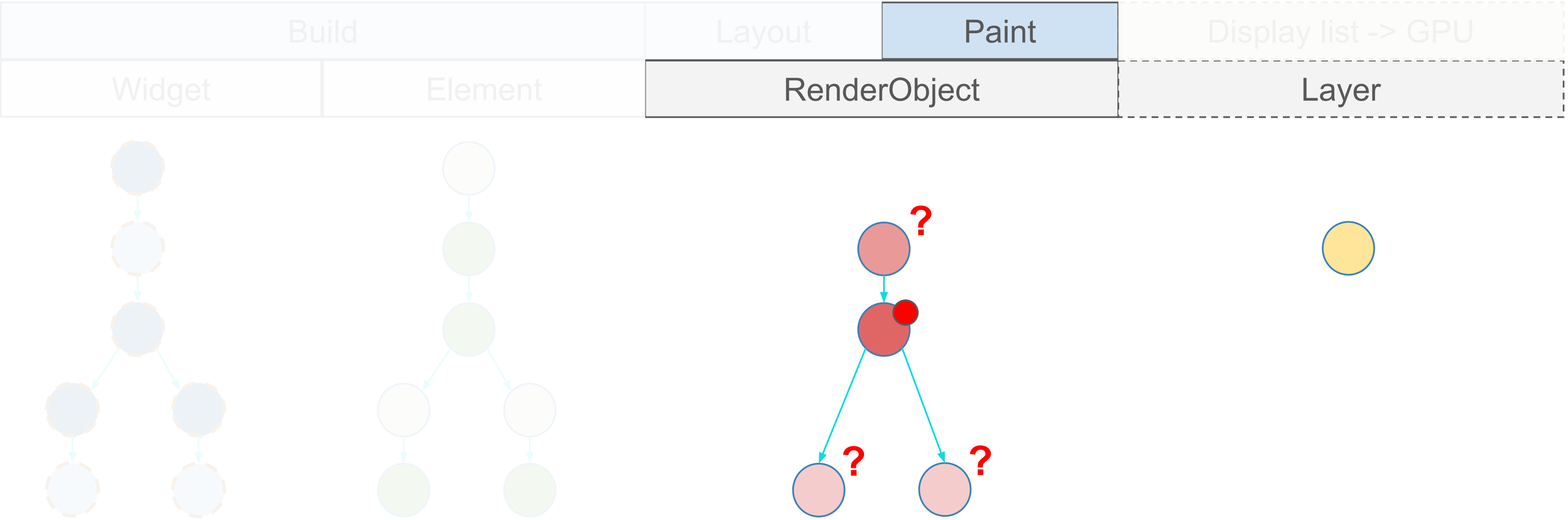


如何提高paint效率

How to Improve Paint Efficiency

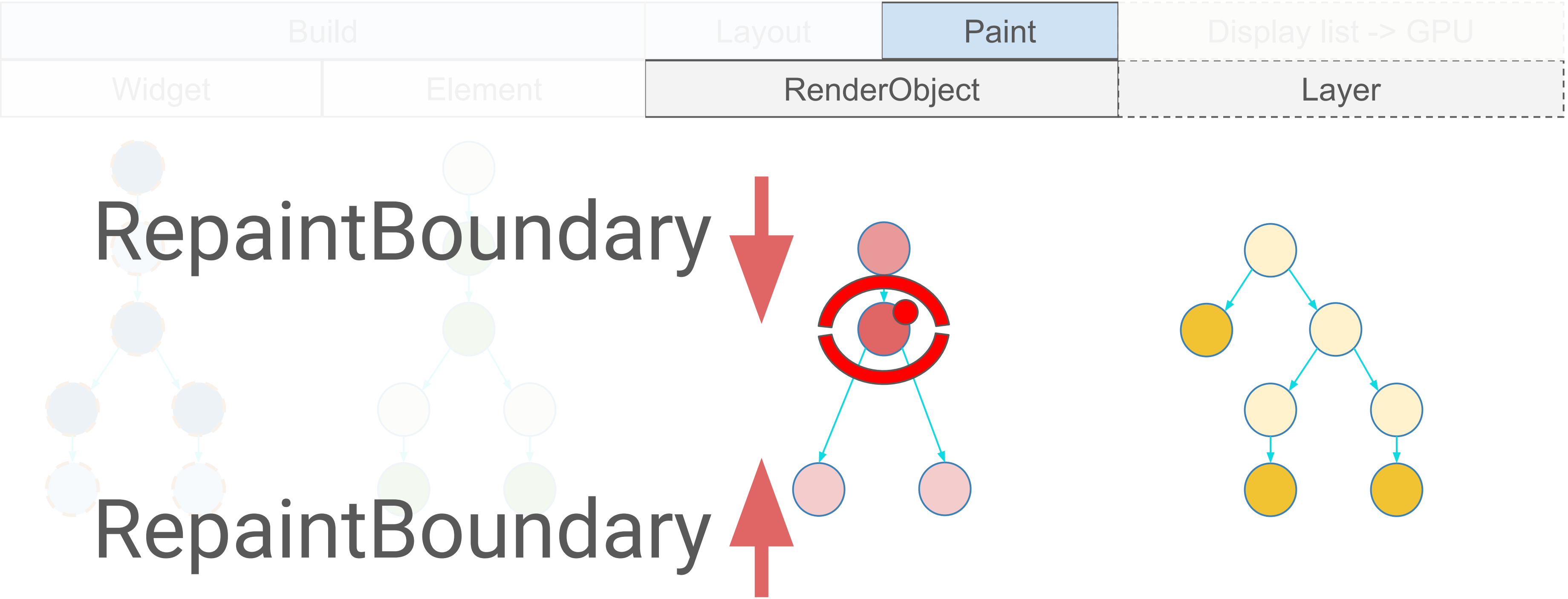
如何提高paint效率

How to Improve Paint Efficiency



如何提高paint效率

How to Improve Paint Efficiency



调试工具

Performance Debugging Tools

- debugDumpLayerTree
  - 查看layer树
- debugPaintLayerBordersEnabled
  - 查看layer界限
- debugRepaintRainbowEnabled
  - 被重新绘制的RenderObject
- debugProfilePaintsEnabled
  - 在观测台里显示绘制树

调试工具

Performance Debugging Tools

- debugDumpLayerTree
  - 查看layer树
- debugPaintLayerBordersEnabled
  - 查看layer界限
- debugRepaintRainbowEnabled
  - 被重新绘制的RenderObject
- debugProfilePaintsEnabled
  - 在了望台里显示绘制树

DEMO

xster@google.com

@xster

Build		Layout	Paint	Display list -> GPU
Widget	Element	RenderObject		Layer