Flutter

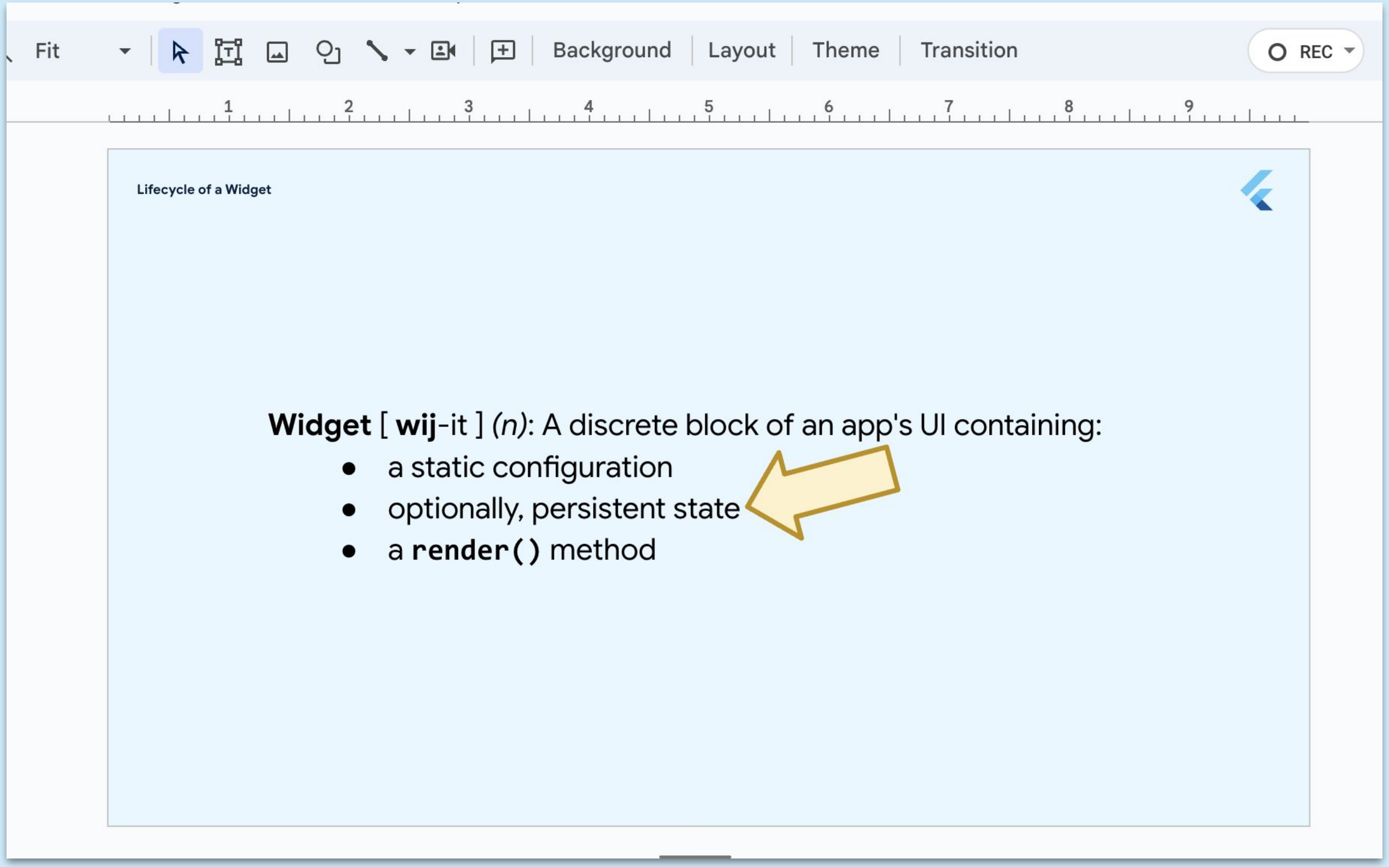
Life cycle of a RenderObject

FlutterConnection, Paris, June 2, 2023

Craig Labenz

Developer Relations Engineer, Google

Life cycle of a RenderObject

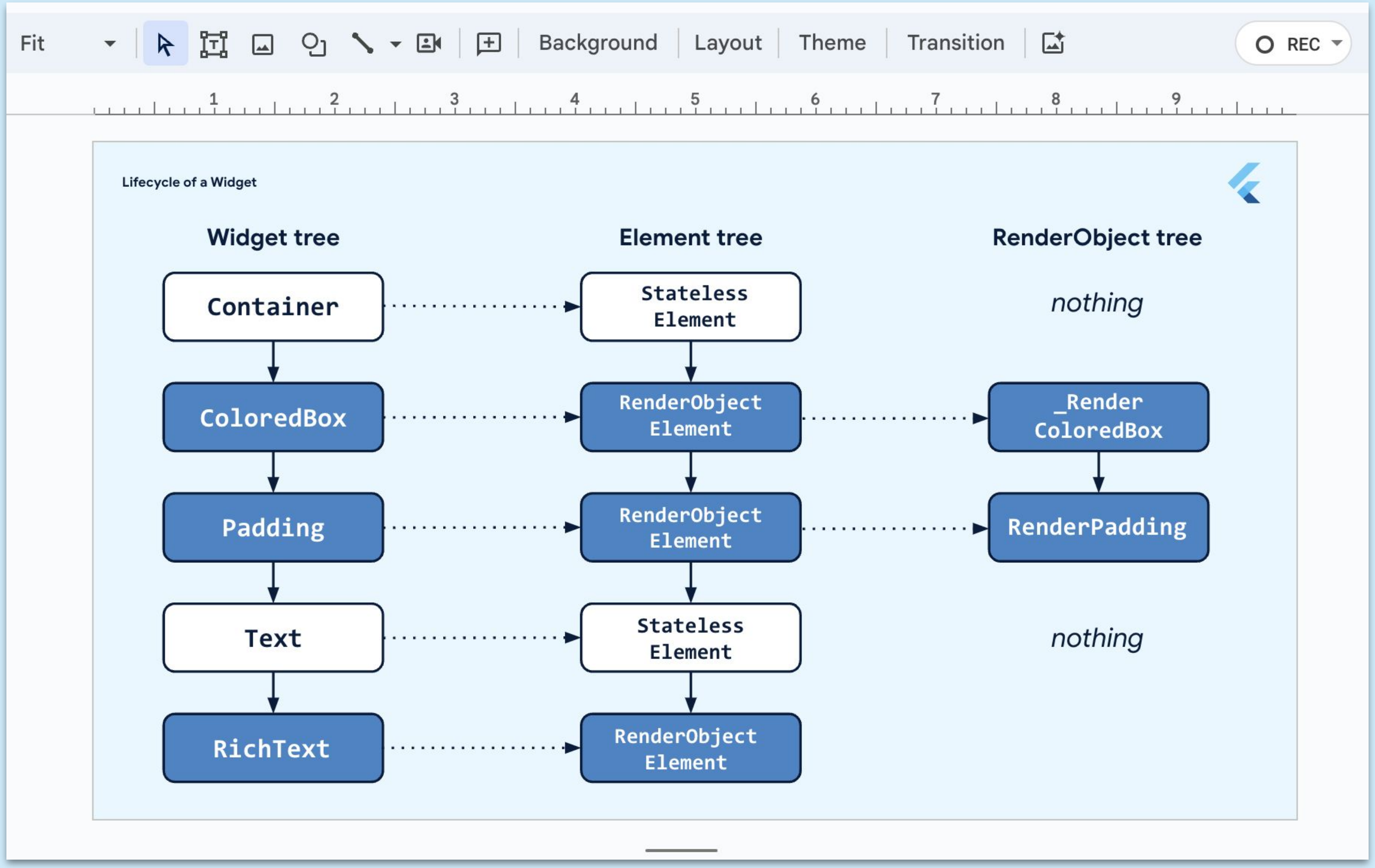


Lifecycle of a Widget

Widget [**wij**-it] *(n)*: A discrete block of an app's UI containing:

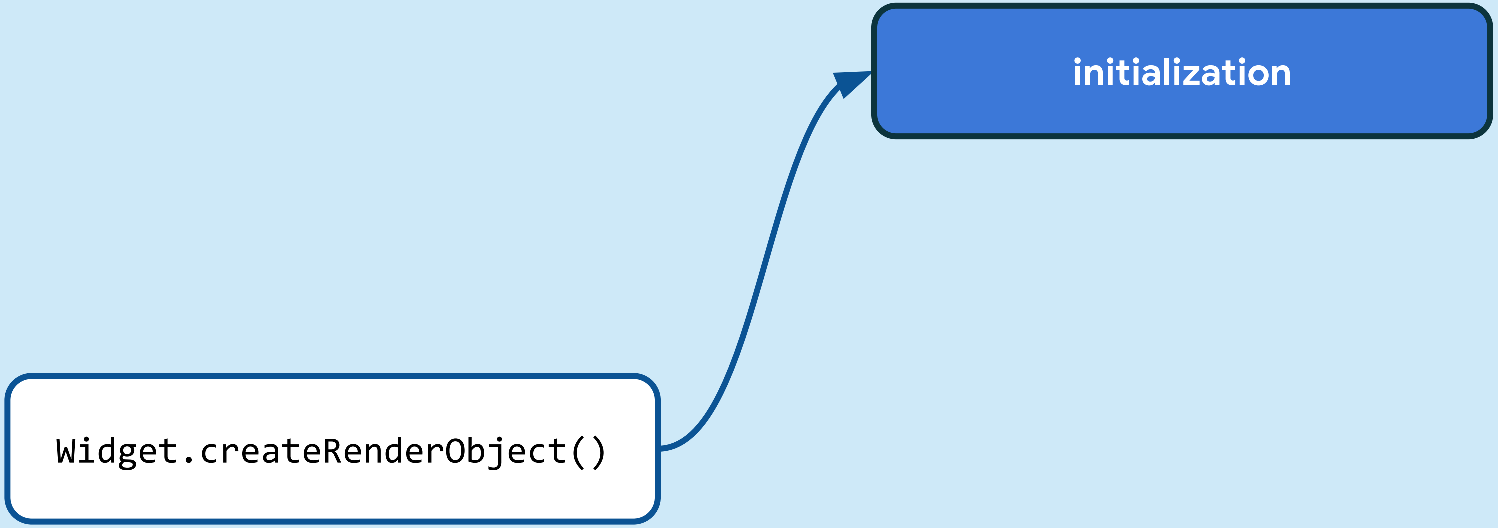
- a static configuration
- optionally, persistent state
- a `render()` method

Life cycle of a RenderObject



RenderObject

Life cycle of a RenderObject

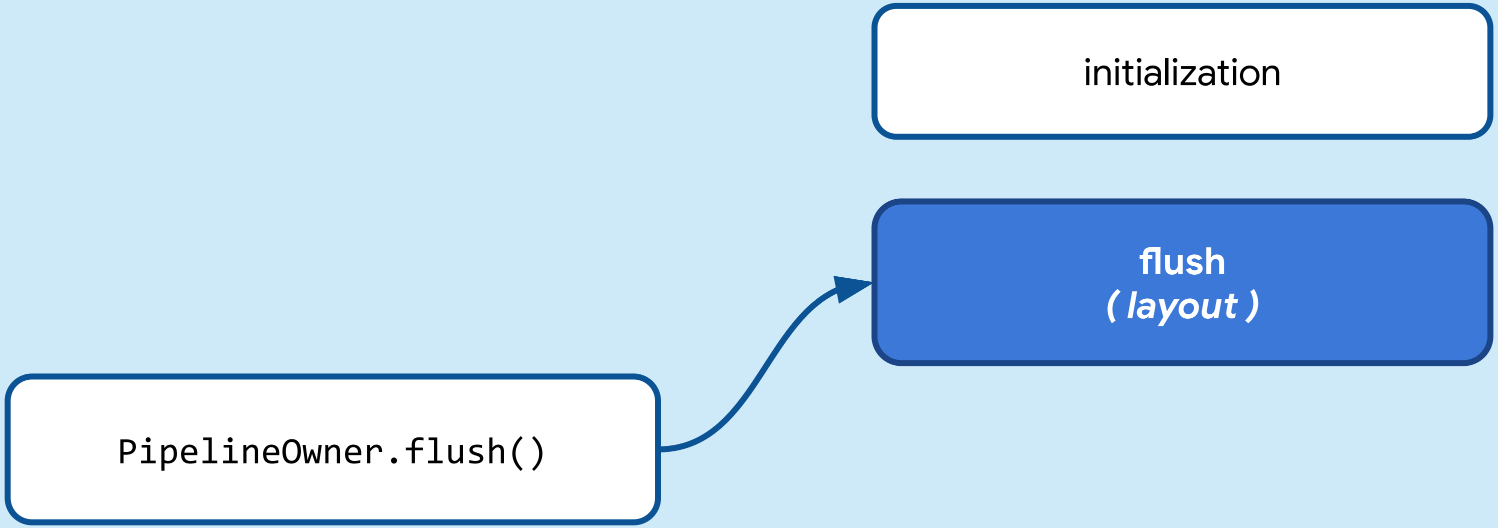


Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {}
```

Layout

Life cycle of a RenderObject



Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void layout(Constraints constraints);
}
```

Life cycle of a RenderObject

```dart
abstract class RenderObject extends AbstractNode {
  void layout(Constraints constraints);
  void performLayout();
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void layout(Constraints constraints);
  void performLayout();
  void markNeedsLayout();
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void layout(Constraints constraints);
  void performLayout();
  void markNeedsLayout();
}
```

Life cycle of a RenderObject

```
abstract class RenderBox extends RenderObject {
  Size? size;

  void performLayout() {
    size = calculateSize();
  }
}
```

Life cycle of a RenderObject

```
class RenderPadding extends RenderShiftedBox {
  void layout(Constraints constraints) {
    _constraints = constraints;
    performLayout();
  }
  void performLayout() {
    final innerConstraints = _constraints - padding;
    child.layout(innerConstraints);
    size = child.size + padding;
  }
}
```

Life cycle of a RenderObject

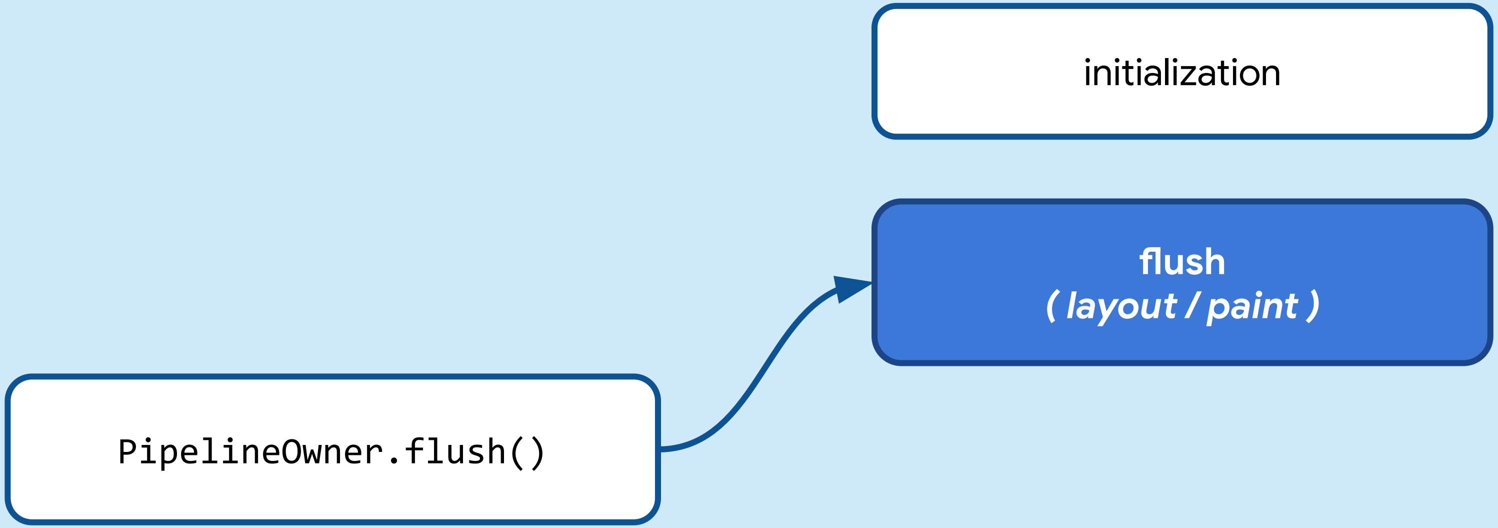
```
class RenderPadding extends RenderShiftedBox {
  void layout(Constraints constraints) {
    _constraints = constraints;
    performLayout();
  }
  void performLayout() {
    final innerConstraints = _constraints - padding;
    child.layout(innerConstraints);
    size = child.size + padding;
  }
}
```

Life cycle of a RenderObject

```
class RenderPadding extends RenderShiftedBox {
  void layout(Constraints constraints) {
    _constraints = constraints;
    performLayout();
  }
  void performLayout() {
    final innerConstraints = _constraints - padding;
    child.layout(innerConstraints);
    size = child.size + padding;
  }
}
```

Painting

Life cycle of a RenderObject



Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void paint(PaintingContext context, Offset offset);
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void paint(PaintingContext context, Offset offset);
  void markNeedsPaint();
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void paint(PaintingContext context, Offset offset);
  void markNeedsPaint();
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void paint(PaintingContext context, Offset offset);
  void markNeedsPaint();
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void paint(PaintingContext context, Offset offset);
  void markNeedsPaint();
}
```

Life cycle of a RenderObject

```
context.canvas.drawRect(size, Paint());
```

Life cycle of a RenderObject

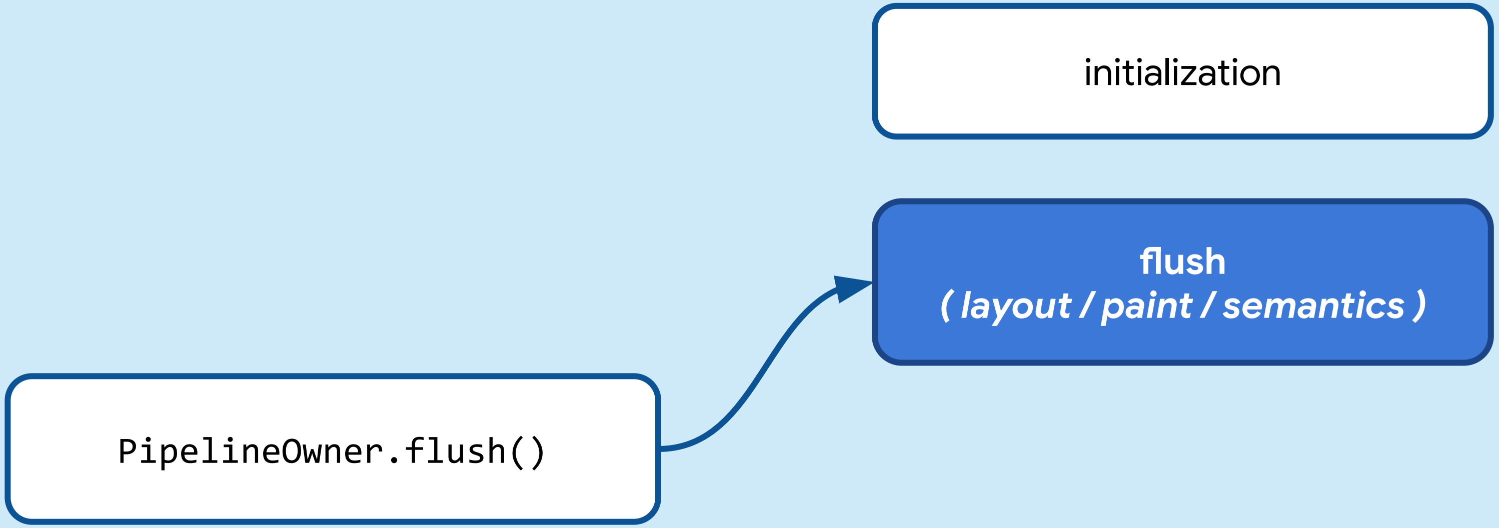
```
// painting.dart

/// An interface for recording graphical operations.
abstract class Canvas {}
```

Life cycle of a RenderObject

```
class RenderTransform extends RenderProxyBox {
  void paint(PaintingContext context, Offset offset) {
    Layer layer = context.pushTransform(...);
  }
}
```

Semantics

Life cycle of a RenderObject



Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void describeSemanticsConfiguration(
    SemanticsConfiguration config,
  );
}
```

Life cycle of a RenderObject

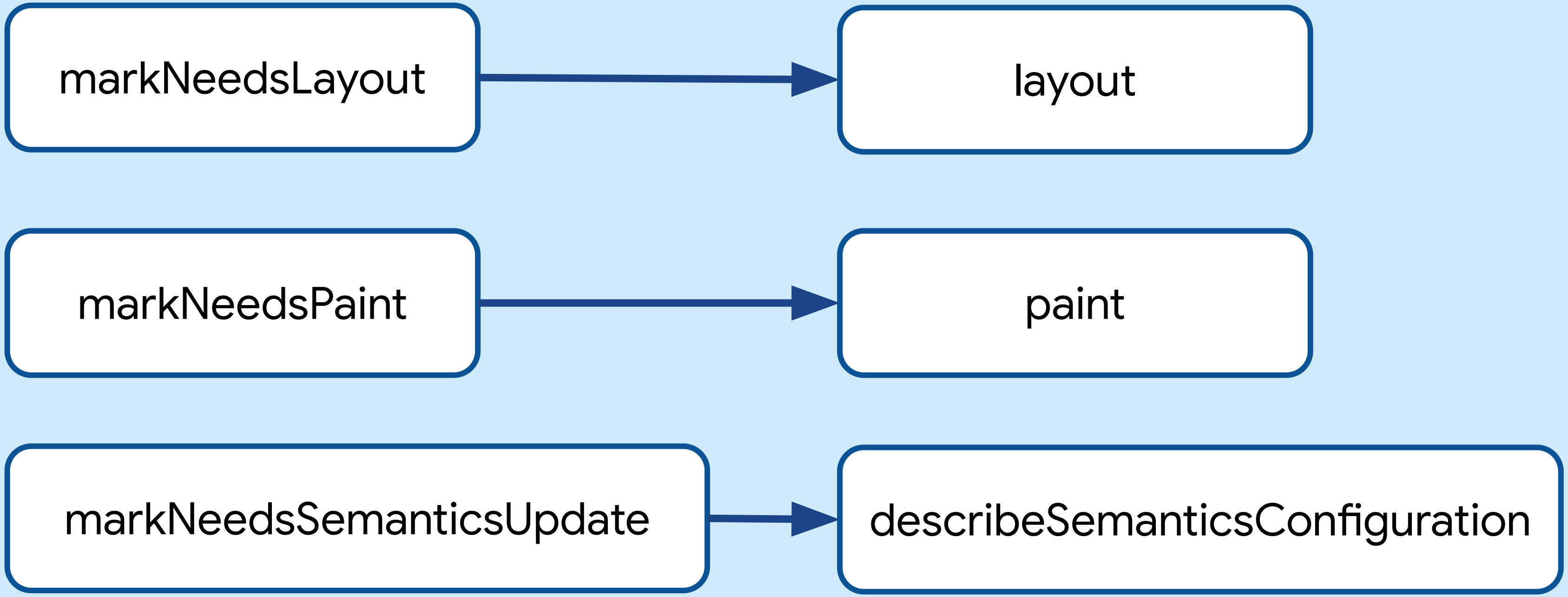
```
abstract class RenderObject extends AbstractNode {
  void describeSemanticsConfiguration(
    SemanticsConfiguration config,
  );
  void markNeedsSemanticsUpdate();
}
```

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void describeSemanticsConfiguration(
    SemanticsConfiguration config,
  ) {
    config.isSemanticBoundary = true;
    config.label = myTextValue;
  }
}
```

Updates

markNeeds*


markNeedsLayout
layout
markNeedsPaint
paint
markNeedsSemanticsUpdate
describeSemanticsConfiguration

Life cycle of a RenderObject

```
class RenderString extends RenderBox {
  RenderString({required String value});
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {
  RenderString({required String value}) : _value = value;

  String _value;
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;



  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;



  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;



  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;
    markNeedsPaint();


  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;
    markNeedsPaint();
    markNeedsLayout();
  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;
    ~~markNeedsPaint();~~
    markNeedsLayout();
  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;
    markNeedsSemanticsUpdate();
    markNeedsLayout();
  }
}
```

Life cycle of a RenderObject

```
void markNeedsLayout() {
  if (parent.isSameLayer) {
    parent.markNeedsLayout();
  }
  owner!._nodesNeedingLayout.add(this);
  owner!.requestVisualUpdate();
}
```

Life cycle of a RenderObject

```
void markNeedsLayout() {
  if (parent.isSameLayer) {
    parent.markNeedsLayout();
  }
  owner!._nodesNeedingLayout.add(this);
  owner!.requestVisualUpdate();
}
```

Life cycle of a RenderObject

```
void markNeedsLayout() {
  if (parent.isSameLayer) {
    parent.markNeedsLayout();
  }
  owner!._nodesNeedingLayout.add(this);
  owner!.requestVisualUpdate();
}
```

Life cycle of a RenderObject

```
void markNeedsLayout() {
  if (parent.isSameLayer) {
    parent.markNeedsLayout();
  }
  owner!._nodesNeedingLayout.add(this);
  owner!.requestVisualUpdate();
}
```

Life cycle of a RenderObject

```dart
class StringWidget extends LeafRenderObjectWidget {
  StringWidget({required this.value});
  final String value;

  void updateRenderObject(RenderString renderObject) {
    renderObject.value = value;
  }
}
```

Life cycle of a RenderObject

```
class StringWidget extends LeafRenderObjectWidget {
  StringWidget({required this.value});
  final String value;

  void updateRenderObject(RenderString renderObject) {
    renderObject.value = value;
  }
}
```

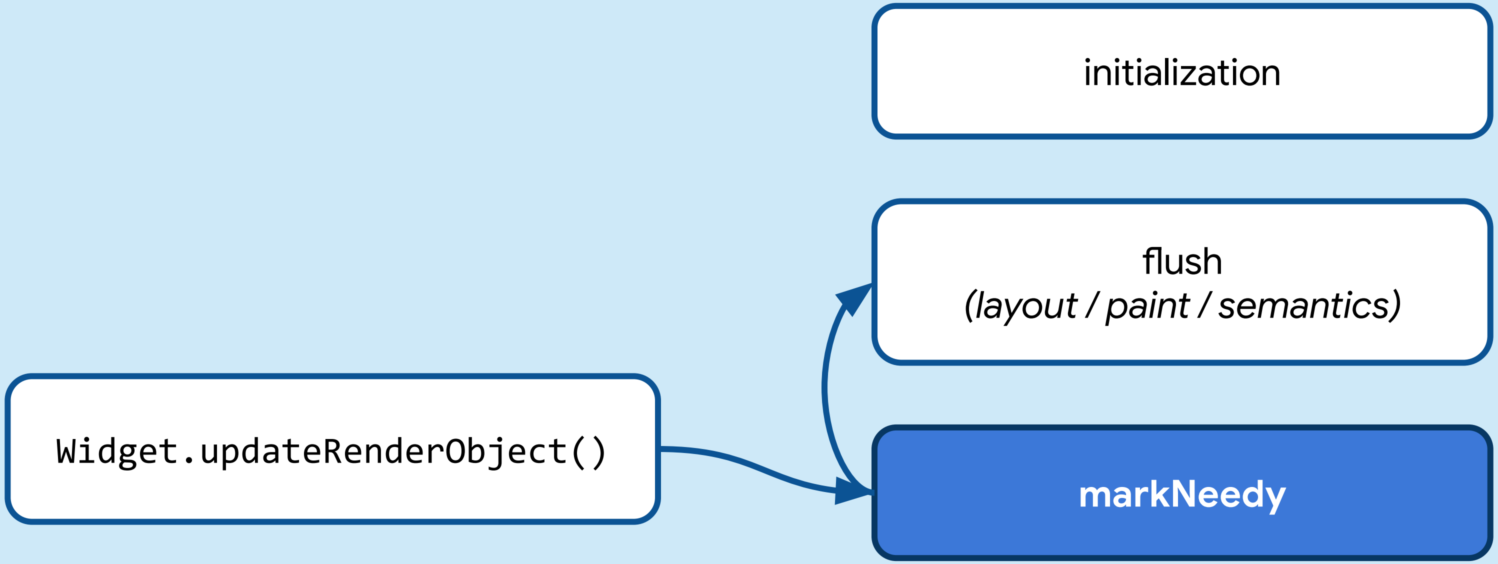
Life cycle of a RenderObject

```
class StringWidget extends LeafRenderObjectWidget {
  StringWidget({required this.value});
  final String value;

  void updateRenderObject(RenderString renderObject) {
    renderObject.value = value;
  }
}
```

Life cycle of a RenderObject

```
class RenderString extends RenderBox {

  void set value(String newValue) {
    if (newValue == _value) return;
    _value = newValue;
    markNeedsSemanticsUpdate();
    markNeedsLayout();
  }
}
```

Life cycle of a RenderObject



Hit testing

Life cycle of a RenderObject

```dart
abstract class RenderBox extends RenderObject {
  bool hitTest(BoxHitTestResult result, {required Offset position}) {
    if (!size.contains(position)) return false;
    if (hitTestChildren(result, position) ||
      hitTestSelf(result, position)) {
        result.add(BoxHitTestEntry(this, position));
        return true;
    }
    return false;
  }
}
```

Life cycle of a RenderObject

```
abstract class RenderBox extends RenderObject {
  bool hitTest(BoxHitTestResult result, {required Offset position}) {
    if (!size.contains(position)) return false;
    if (hitTestChildren(result, position) ||
      hitTestSelf(result, position)) {
        result.add(BoxHitTestEntry(this, position));
        return true;
    }
    return false;
  }
}
```

Life cycle of a RenderObject

```
abstract class RenderBox extends RenderObject {
  bool hitTest(BoxHitTestResult result, {required Offset position}) {
    if (!size.contains(position)) return false;
    if (hitTestChildren(result, position) ||
      hitTestSelf(result, position)) {
        result.add(BoxHitTestEntry(this, position));
        return true;
    }
    return false;
  }
}
```

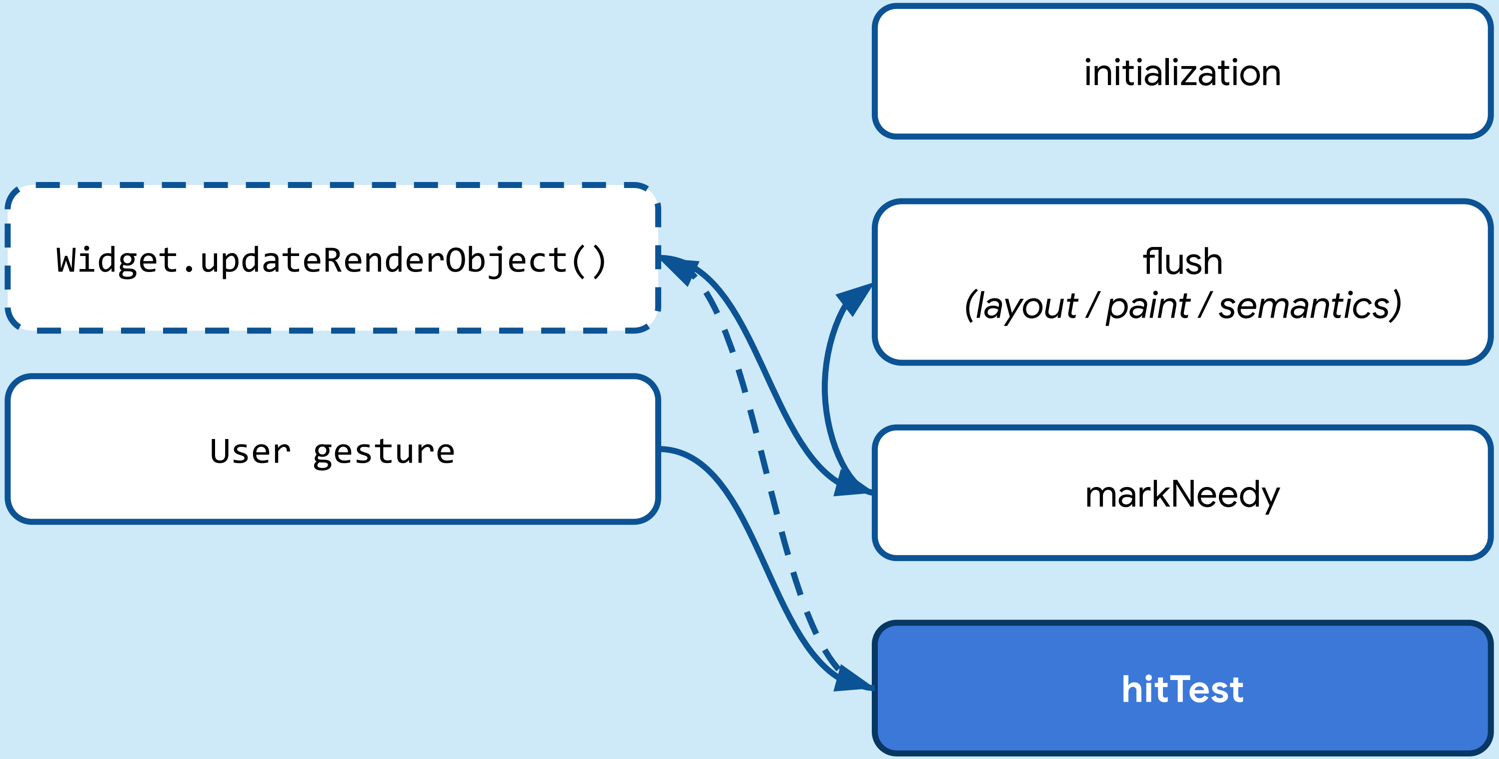
Life cycle of a RenderObject

```dart
abstract class RenderBox extends RenderObject {
  bool hitTest(BoxHitTestResult result, {required Offset position}) {
    if (!size.contains(position)) return false;
    if (hitTestChildren(result, position) ||
      hitTestSelf(result, position)) {
        result.add(BoxHitTestEntry(this, position));
        return true;
    }
    return false;
  }
}
```

Life cycle of a RenderObject

```
abstract class RenderBox extends RenderObject {
  bool hitTest(BoxHitTestResult result, {required Offset position}) {
    if (!size.contains(position)) return false;
    if (hitTestChildren(result, position) ||
      hitTestSelf(result, position)) {
        result.add(BoxHitTestEntry(this, position));
        return true;
    }
    return false;
  }
}
```

Life cycle of a RenderObject

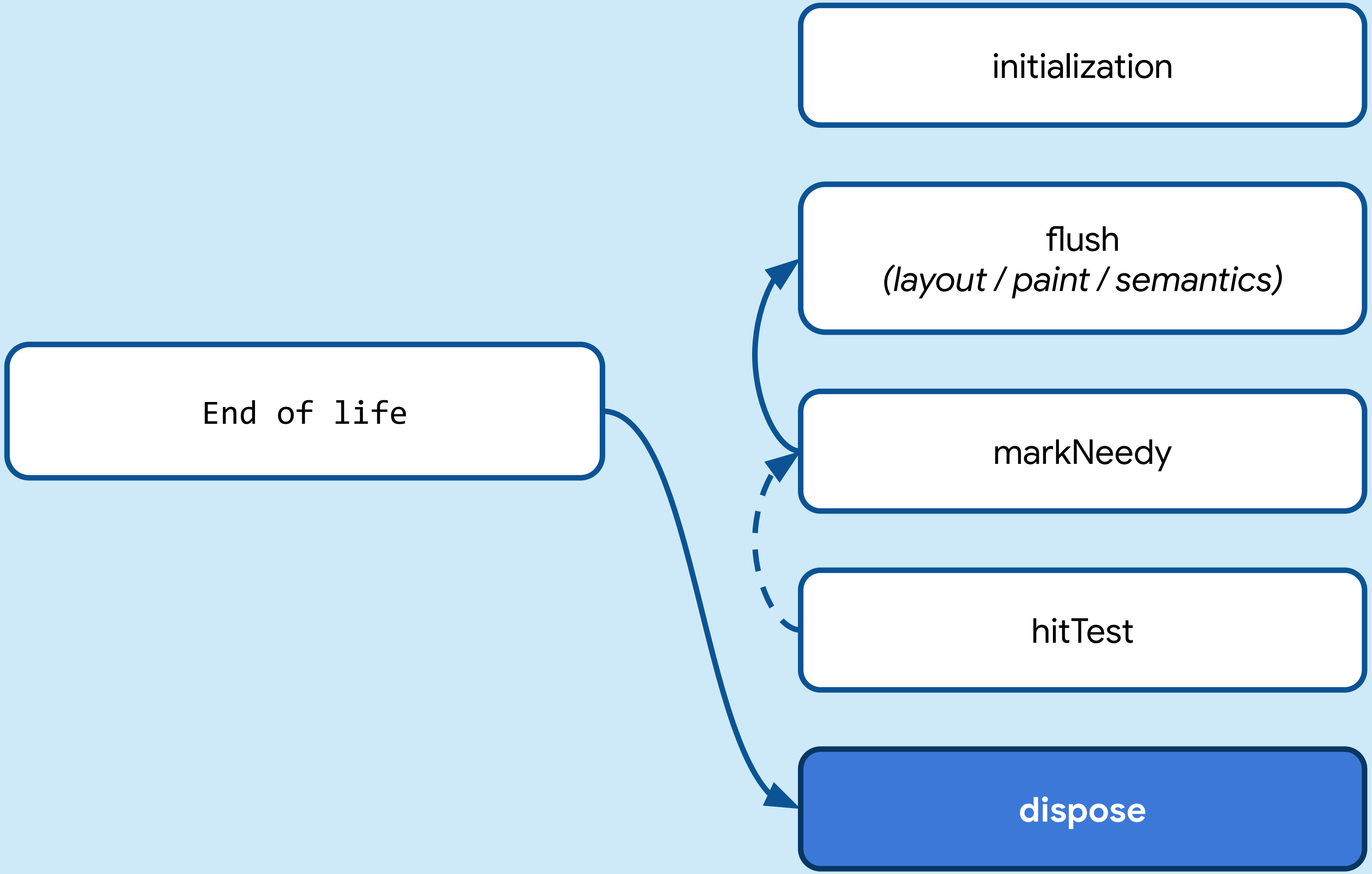


Dispose

Life cycle of a RenderObject

```
abstract class RenderObject extends AbstractNode {
  void dispose() {
    layer?.dispose();
    textPainter?.dispose();
    super.dispose();
  }
}
```

Life cycle of a RenderObject



ParentData

Life cycle of a RenderObject

```
class RenderFlex extends RenderBox {

  void addAll(List<Widget> children) {
    Widget? previousChild;
    for (final child in children) {
      child.parentData = FlexParentData();
      previousChild?.parentData.nextSibling = child;
      previousChild = child;
    }
  }
}
```

Life cycle of a RenderObject

Stack :: Positioned

</ RenderObject>

</ RenderObject>

The Layer Tree

Life cycle of a RenderObject

```
class RenderTransform extends RenderProxyBox {
  void paint(PaintingContext context, Offset offset) {
    context.canvas.drawRect(size, Paint());
  }
}
```

Life cycle of a RenderObject

```
class RenderTransform extends RenderProxyBox {
  void paint(PaintingContext context, Offset offset) {
    context.canvas.drawRect(size, Paint());
  }
}
```

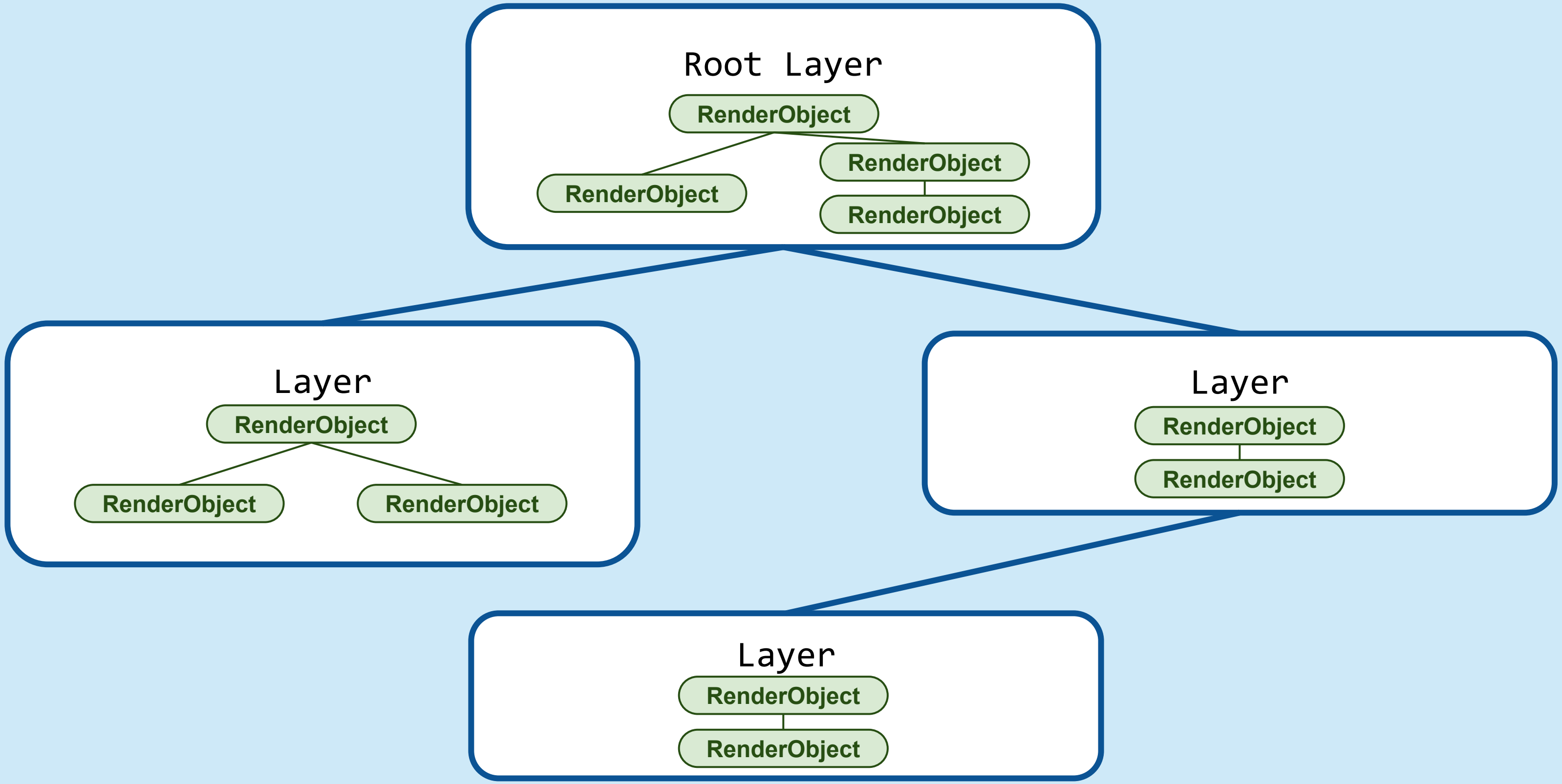
Life cycle of a RenderObject

```
class RenderTransform extends RenderProxyBox {
  void paint(PaintingContext context, Offset offset) {
    context.canvas.drawRect(size, Paint());
  }
}
```

Life cycle of a RenderObject

```
class RenderTransform extends RenderProxyBox {
  void paint(PaintingContext context, Offset offset) {
    layer = context.pushTransform(...);
  }
}
```

Life cycle of a RenderObject



Layers

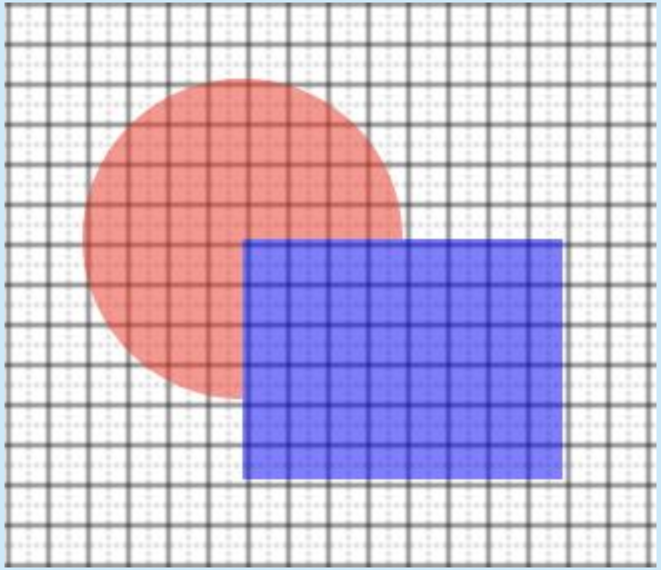
Life cycle of a RenderObject

```
10 *  3 + 5  = 35

10 * (3 + 5) = 80
```

Life cycle of a RenderObject

```
Background + Offset + Opacity(Circle + Square)
```

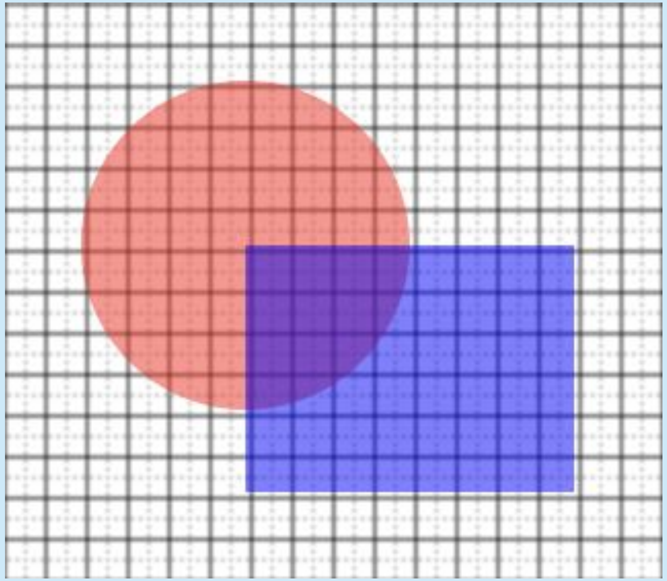
`Background + Offset + Opacity(Circle + Square)`



Life cycle of a RenderObject

```
Background + Offset + Opacity(Circle) + Opacity(Square)
```

`Background + Offset + Opacity(Circle) + Opacity(Square)`



Needing Layers is rare

</ Layers>

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.Layer rootLayer;
  ui.SceneBuilder builder;
  ui.FlutterView view;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.Layer rootLayer;
  ui.SceneBuilder builder;
  ui.FlutterView view;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.Layer rootLayer;
  ui.SceneBuilder builder;
  ui.FlutterView view;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.Layer rootLayer;
  ui.SceneBuilder builder;
  ui.FlutterView view;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.Layer rootLayer;
  ui.SceneBuilder builder;
  ui.FlutterView view;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.FlutterView view;
  ui.Layer rootLayer;
  ui.SceneBuilder builder;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

Life cycle of a RenderObject

```
class WidgetsFlutterBinding with RenderBinding {
  ui.FlutterView view;
  ui.Layer rootLayer;
  ui.SceneBuilder builder;

  void drawFrame() {
    pipelineOwner.flush();
    final ui.Scene scene = builder.build(rootLayer);
    view.render(scene);
  }
}
```

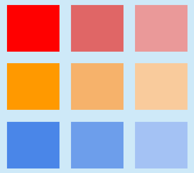
Shader A ➡️

Shader B ➡️

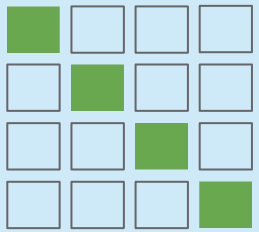
Shader C ➡️

Life cycle of a RenderObject

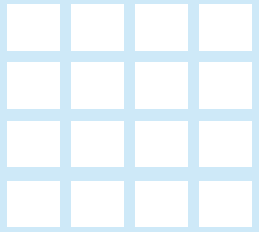
Shader A ➡️

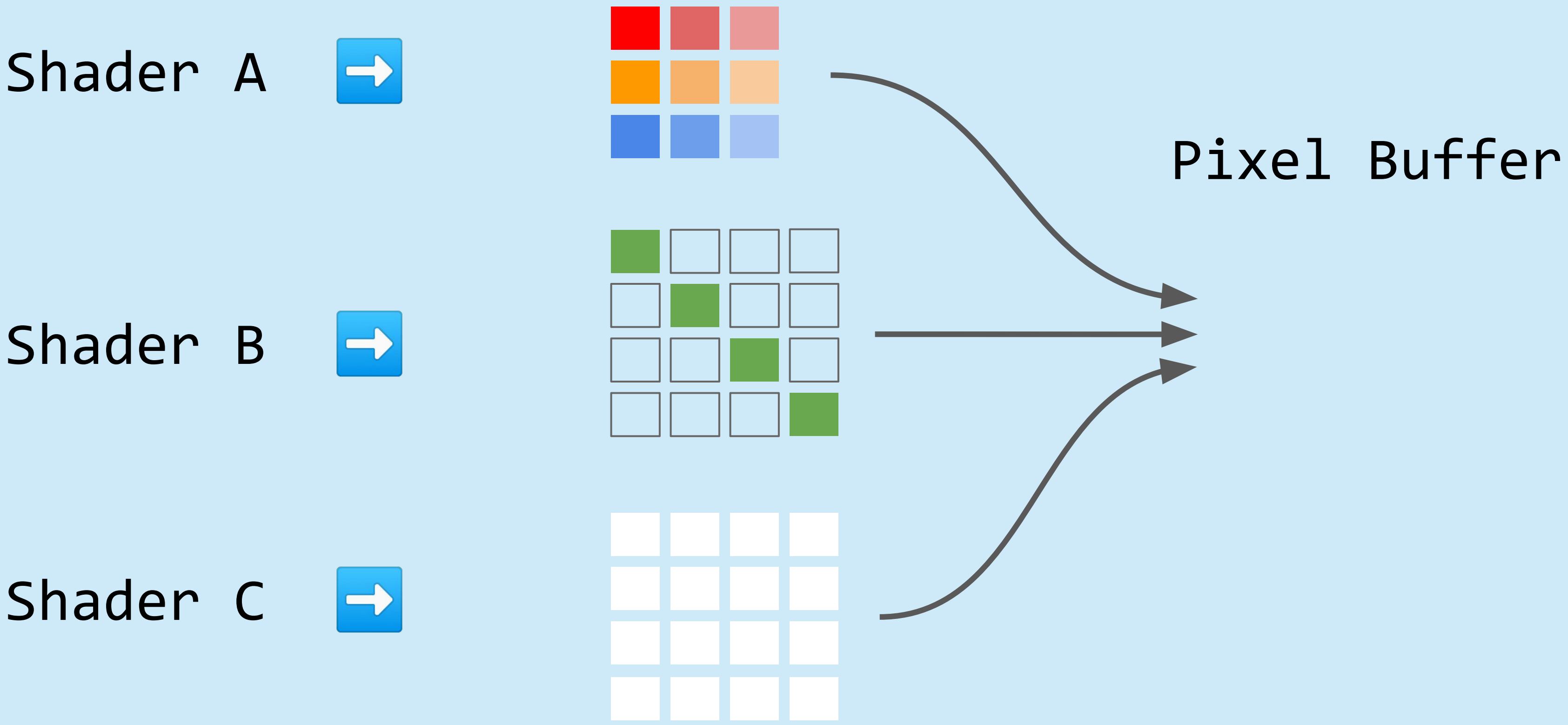


Shader B ➡️

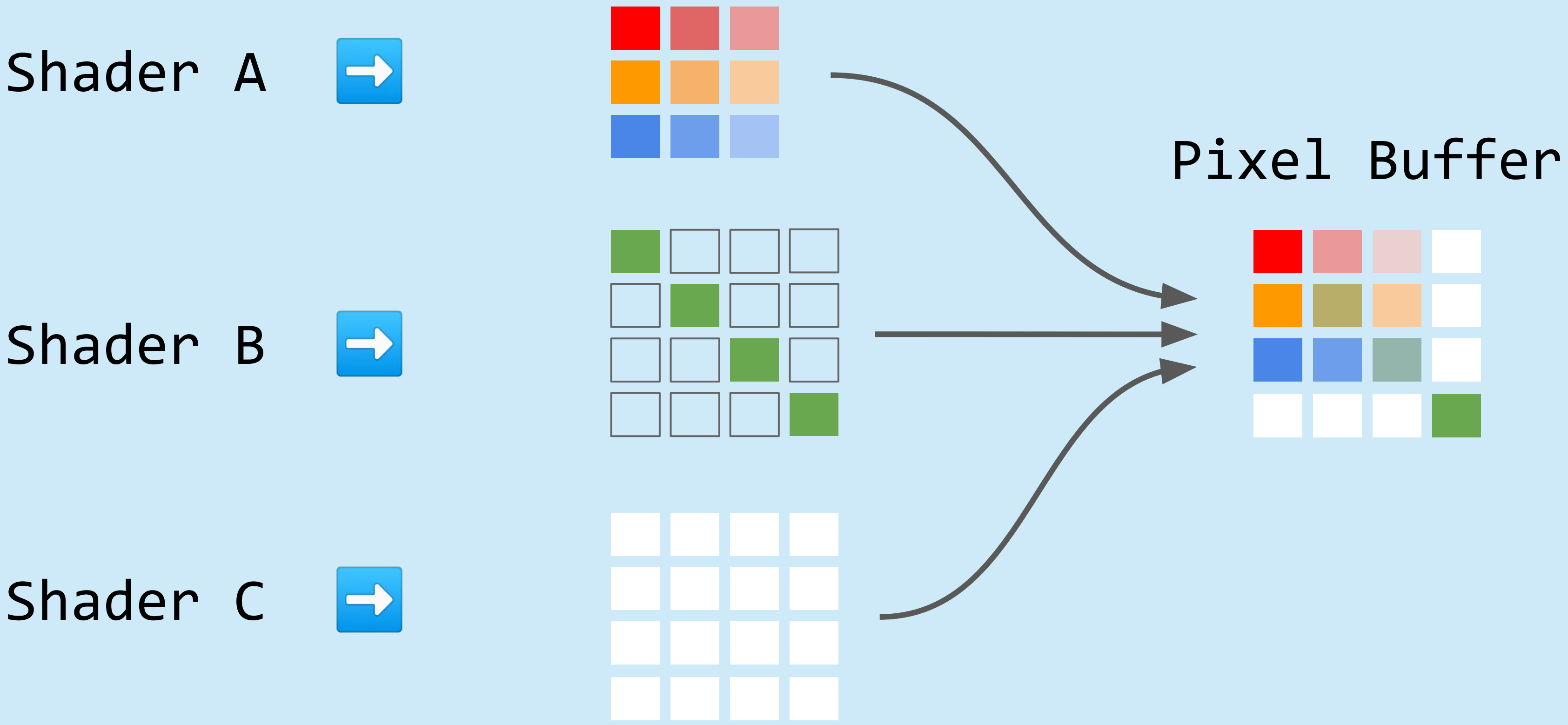


Shader C ➡️




Shader A ➡️
Shader B ➡️
Shader C ➡️
Pixel Buffer

Life cycle of a RenderObject

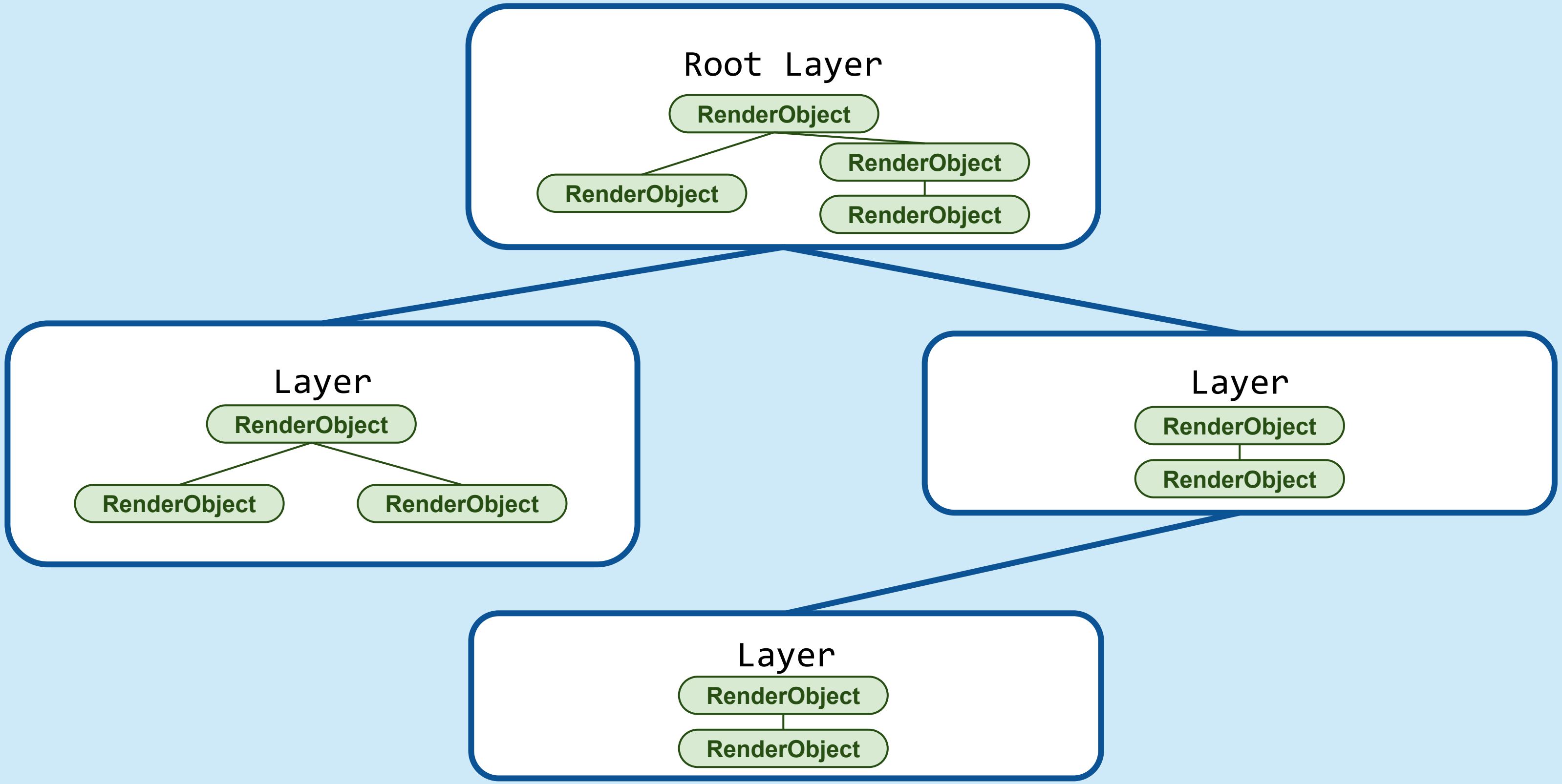


Pixels on the screen!

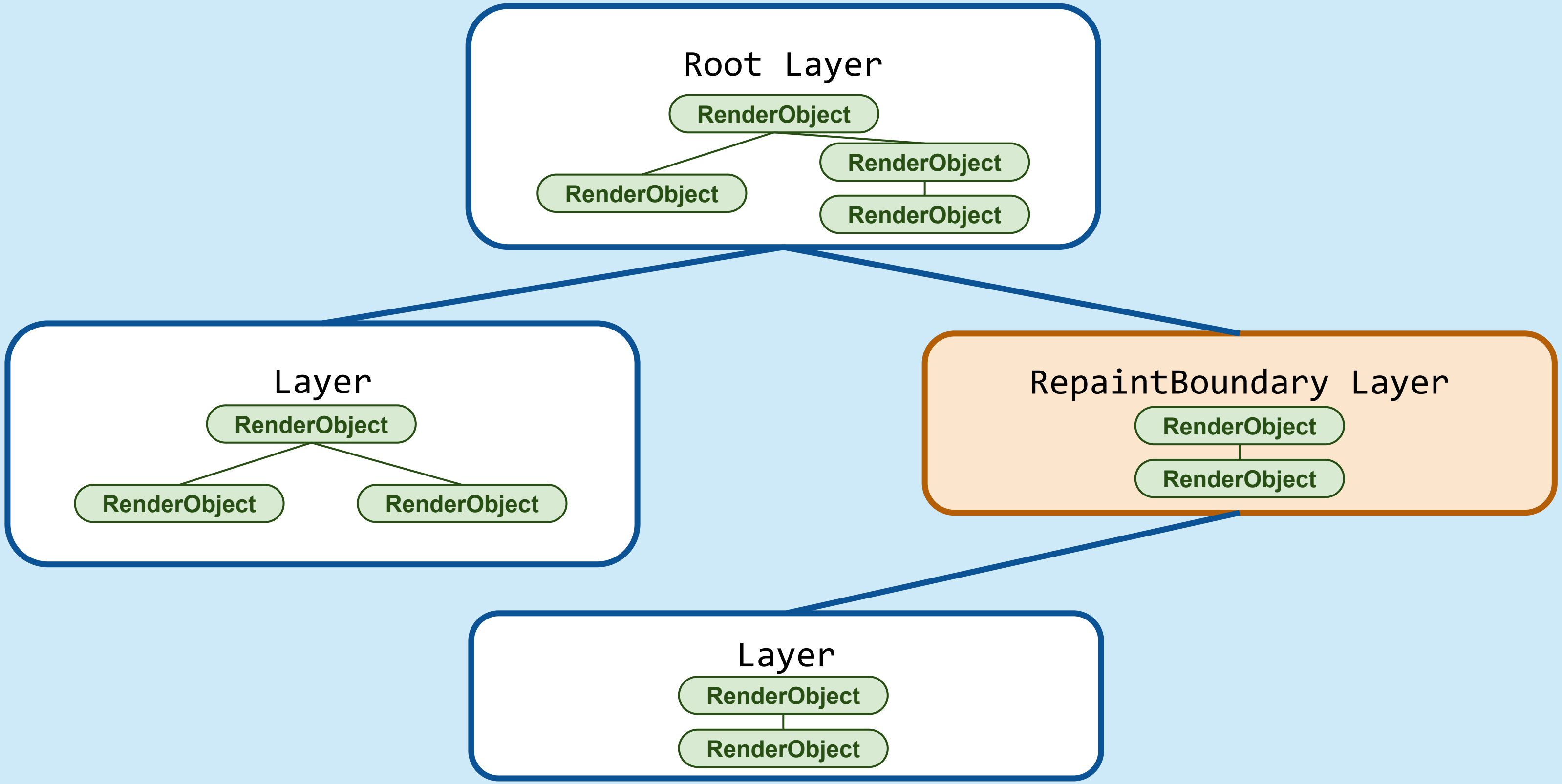
1. What happens when nothing is happening?

2. RepaintBoundary

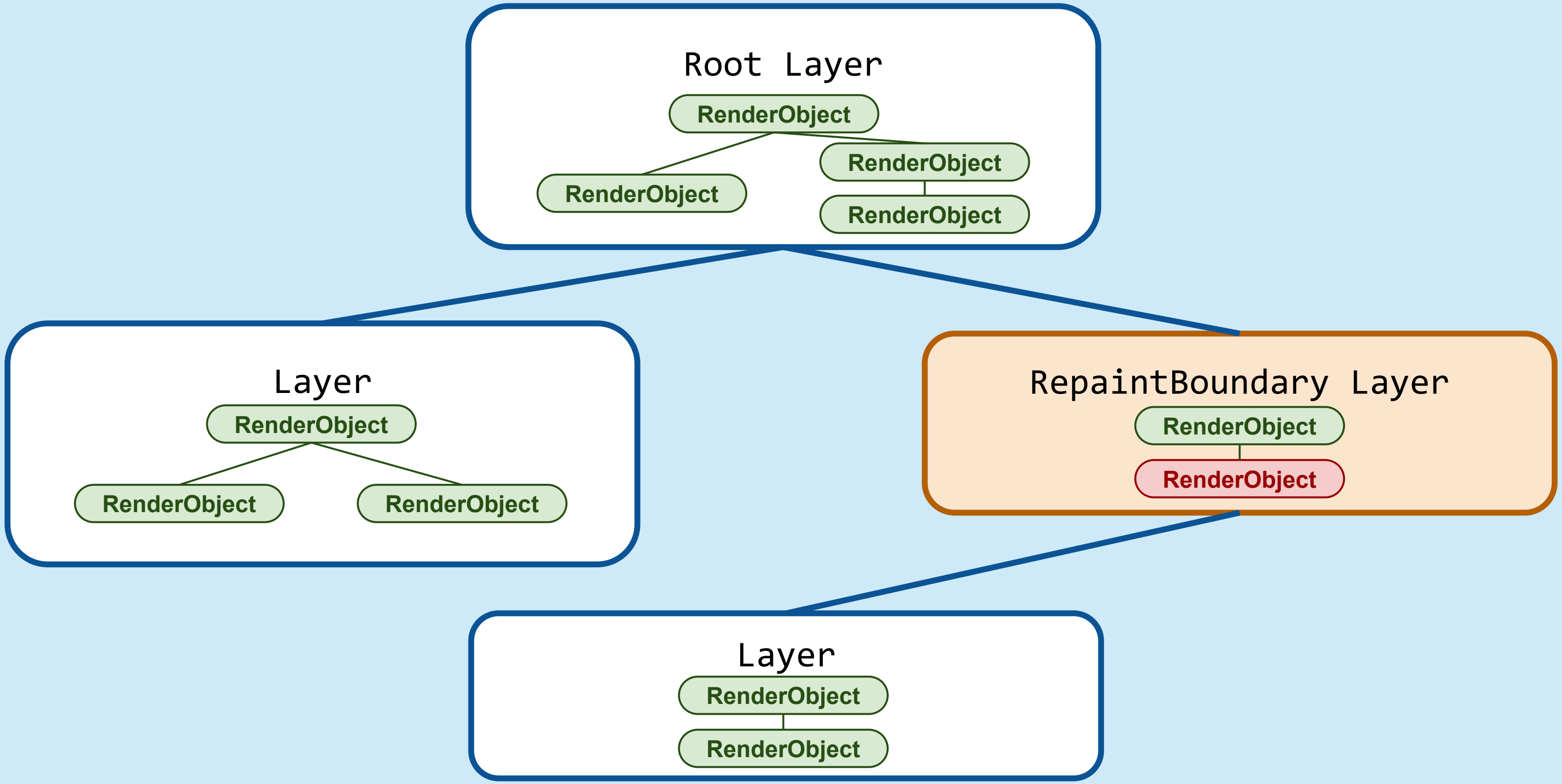
Life cycle of a RenderObject



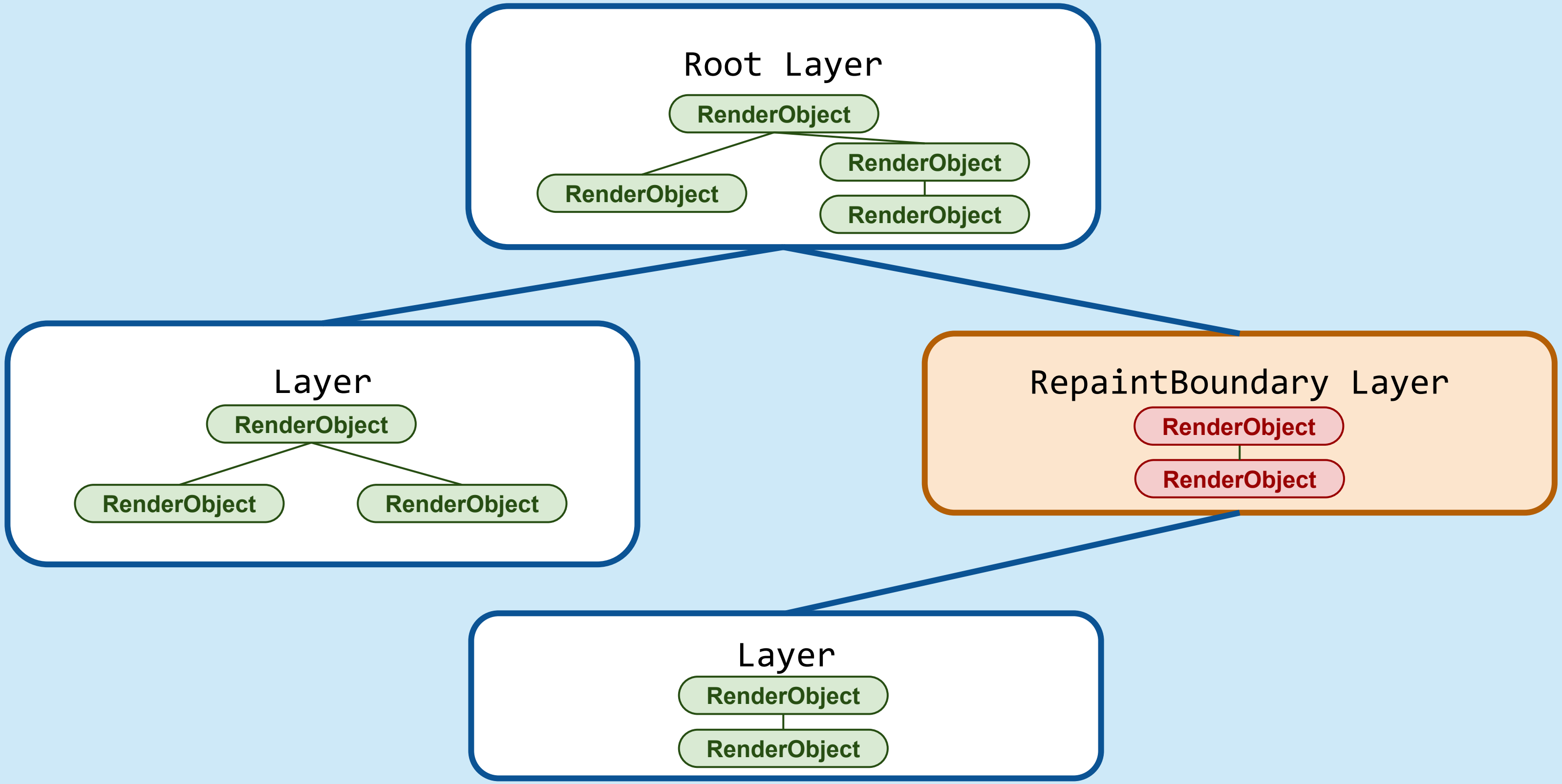
Life cycle of a RenderObject



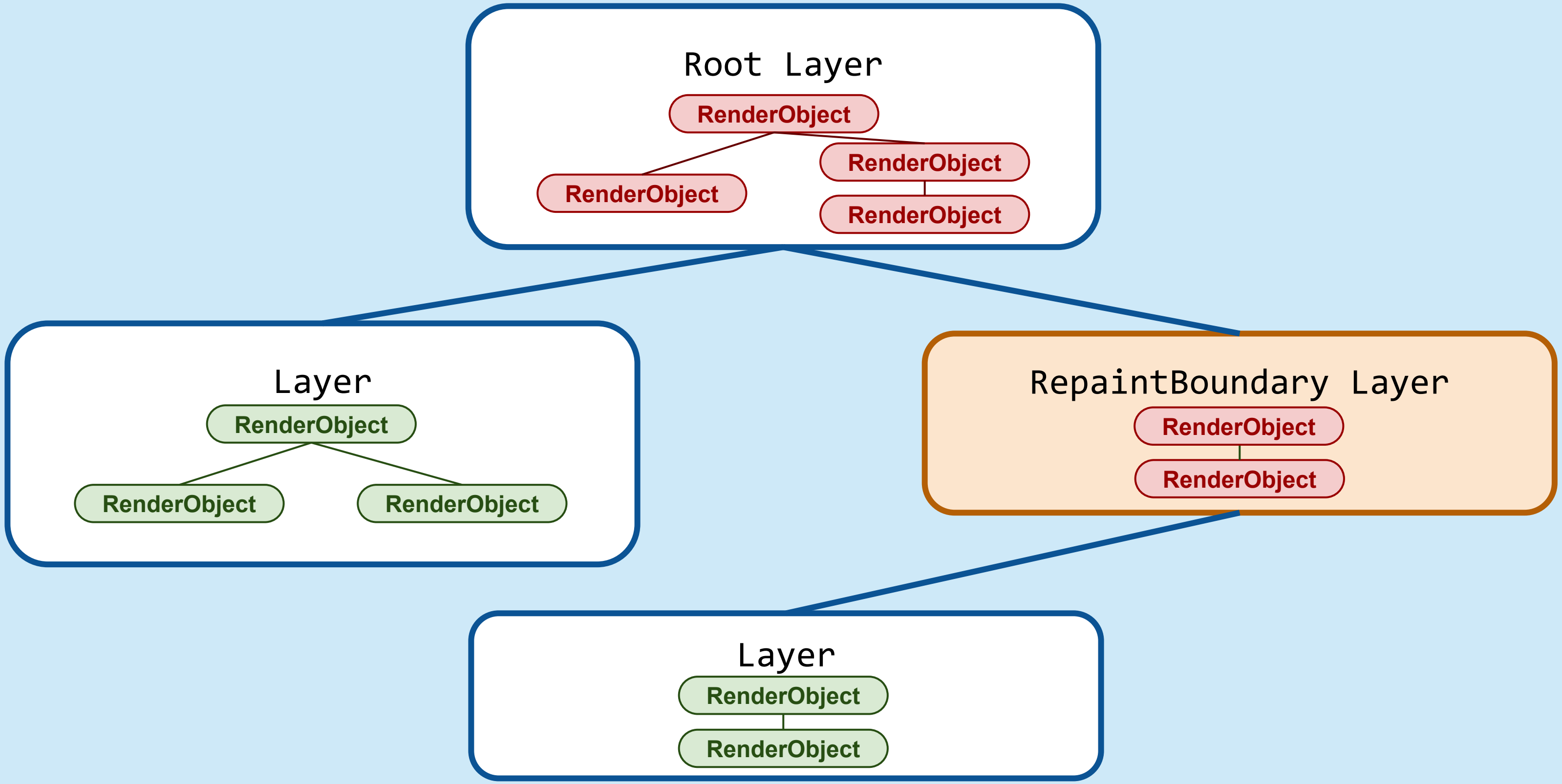
Life cycle of a RenderObject



Life cycle of a RenderObject



Life cycle of a RenderObject



Scrolling items

Scrolling item is animating

Life cycle of a RenderObject

```
GestureDetector(
  onTap: () => setState(() => myColor = getNextColor()),
  child: ColoredBox(
    color: myColor
    child: RepaintBoundary(
      child: ColoredBox(color: myColor, child: ...),
    ),
  ),
)
```

Life cycle of a RenderObject

```
GestureDetector(
  onTap: () => setState(() => myColor = getNextColor()),
  child: ColoredBox(
    color: myColor
    child: RepaintBoundary(
      child: ColoredBox(color: myColor, child: ...),
    ),
  ),
)
```

Life cycle of a RenderObject

```
GestureDetector(
  onTap: () => setState(() => myColor = getNextColor()),
  child: ColoredBox(
    color: myColor
    child: RepaintBoundary(
      child: ColoredBox(color: myColor, child: ...),
    ),
  ),
)
```

Life cycle of a RenderObject

```
GestureDetector(
  onTap: () => setState(() => myColor = getNextColor()),
  child: ColoredBox(
    color: myColor
    child: RepaintBoundary(
      child: ColoredBox(color: myColor, child: ...),
    ),
  ),
)
```

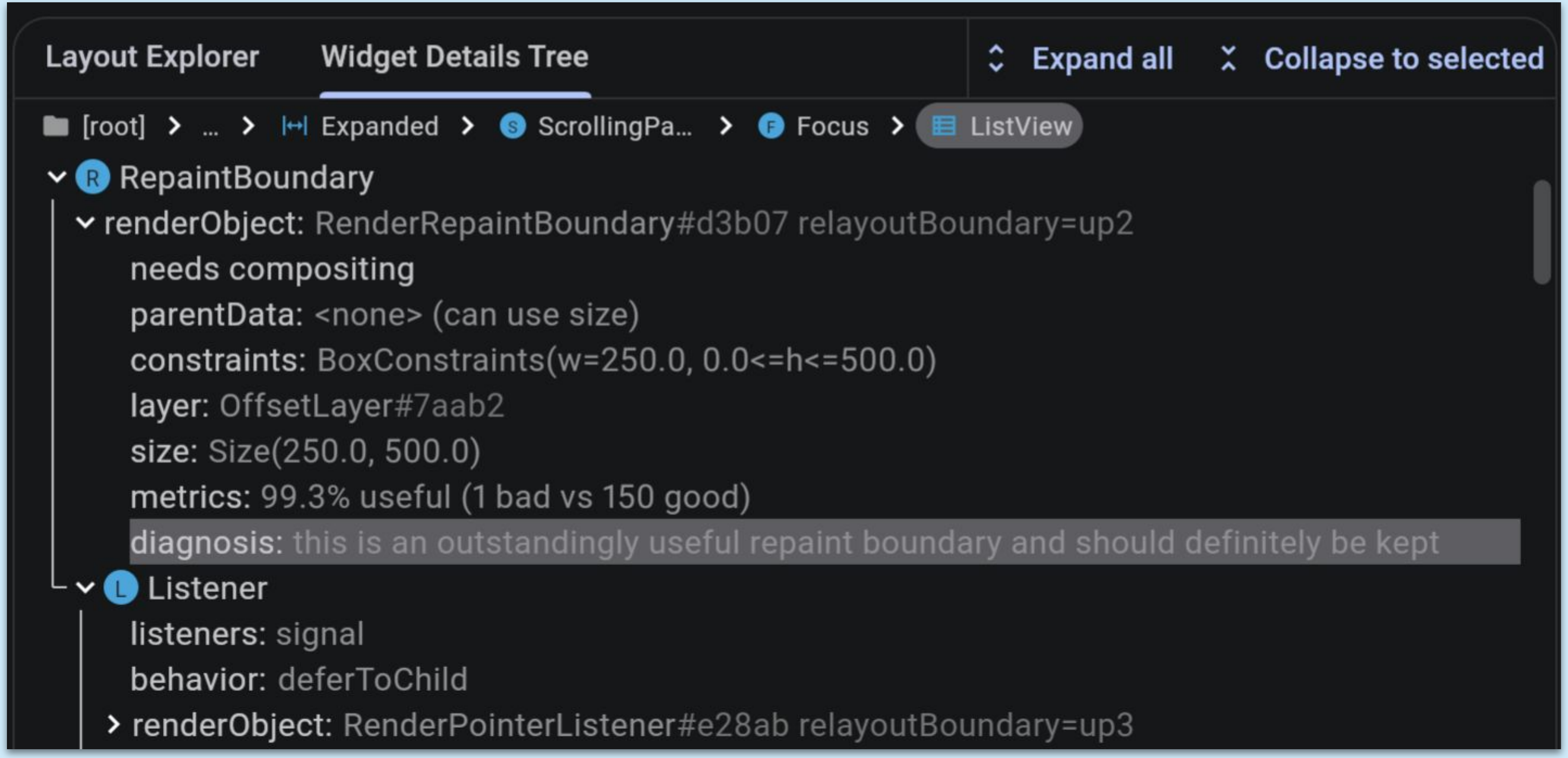
Life cycle of a RenderObject

```
GestureDetector(
  onTap: () => setState(() => myColor = getNextColor()),
  child: ColoredBox(
    color: myColor
    child: RepaintBoundary(
      child: ColoredBox(color: myColor, child: ...),
    ),
  ),
)
```

Life cycle of a RenderObject

```
GestureDetector(
  onTap: () => setState(() => myColor = getNextColor()),
  child: ColoredBox(
    color: myColor
    child: RepaintBoundary(
      child: ColoredBox(color: myColor, child: ...),
    ),
  ),
)
```

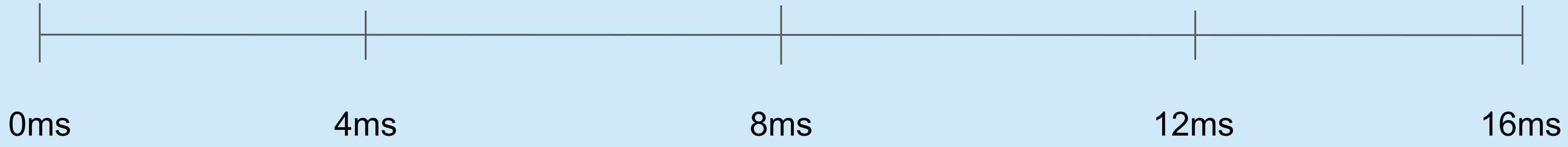
Life cycle of a RenderObject



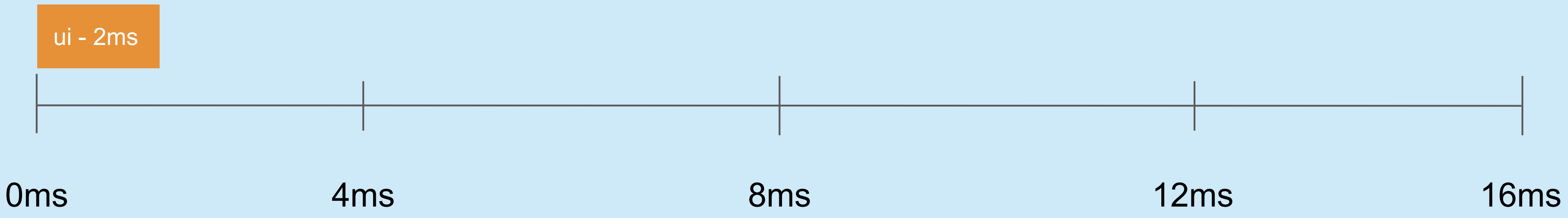
```
RepaintBoundary
  renderObject: RenderRepaintBoundary#d3b07 relayoutBoundary=up2
    needs compositing
    parentData: <none> (can use size)
    constraints: BoxConstraints(w=250.0, 0.0<=h<=500.0)
    layer: OffsetLayer#7aab2
    size: Size(250.0, 500.0)
    metrics: 99.3% useful (1 bad vs 150 good)
    diagnosis: this is an outstandingly useful repaint boundary and should definitely be kept
  Listener
    listeners: signal
    behavior: deferToChild
    renderObject: RenderPointerListener#e28ab relayoutBoundary=up3
```

3. What does this all mean for performance profiling?

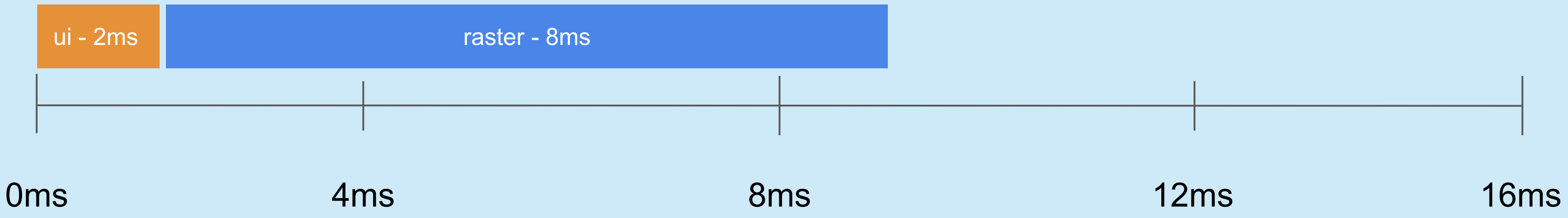
Life cycle of a RenderObject



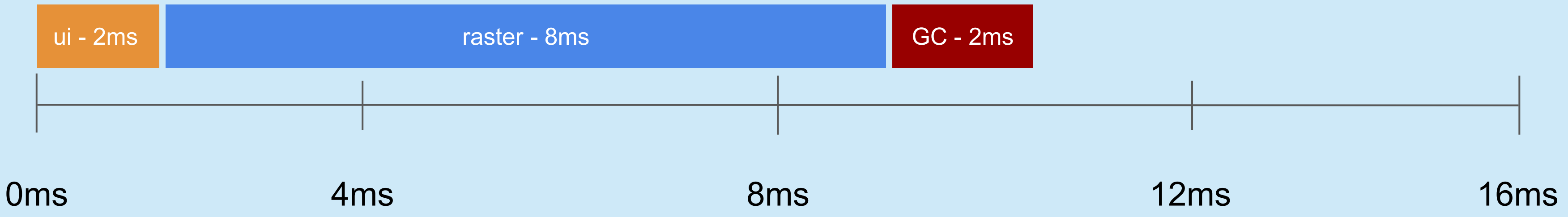
Life cycle of a RenderObject



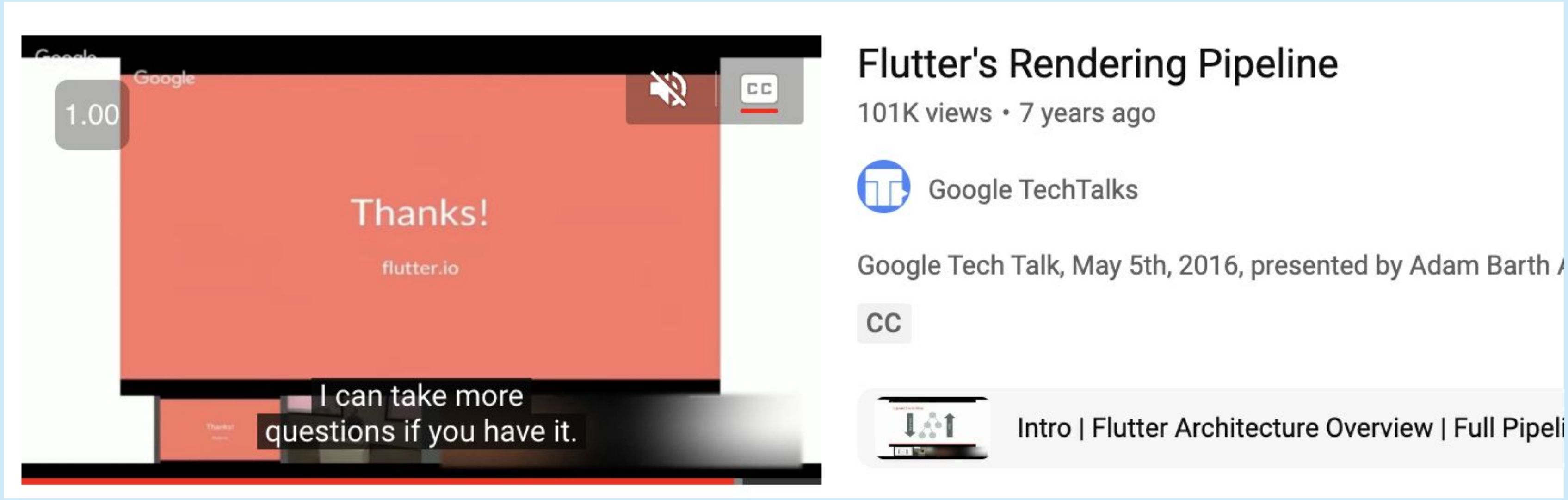
Life cycle of a RenderObject



Life cycle of a RenderObject



https://www.youtube.com/watch?v=UUfXWzp0-DU



I can take more questions if you have it.

Flutter's Rendering Pipeline

101K views • 7 years ago

Google TechTalks

Google Tech Talk, May 5th, 2016, presented by Adam Barth

CC

Intro | Flutter Architecture Overview | Full Pipeli

Special thanks!

Dan Field

Jonah Williams

Michael Goderbauer

Thank you!